

## 開示の概要

情報処理装置(information processing apparatus)は、マンマシンインタフェース、半導体メモリおよび情報処理手段(information processor)を備え、半導体メモリは、アプリケーションソフトウェアコンテンツ部、アプリケーションソフトウェアエンジン、オペレーティングシステム、情報処理手段内ハードウェアドライバ、マンマシンインタフェースドライバからなり、ソフトウェアおよびハードウェアを含む各モジュール間のインタフェースが明確に定義付けられている。

## 発明の名称

家庭用テレビ装置に接続して用いる情報処理装置

Information Processing Apparatus Connected to Home TV Set in Use

## 発明の背景

### 発明の分野

この発明は、家庭用テレビ装置に接続して用いる情報処理装置に関する。より特定のには、たとえば、テレビゲーム機器、通信ネットワーク情報機器、通信カラオケ用機器、カーナビゲーション機器、知育玩具、ワードプロセサ、実用情報提供機器、工場の生産ラインなどで用いられる検査用機器、あるいは各種測定器として用いられる、家庭用テレビ装置に接続して用いる情報処理装置に関する。

### 従来技術の説明

家庭用テレビ装置に接続して使用する情報処理装置は広く普及している。その具体例としては、Super Famicom（登録商標）や Playstation（登録商標）などの家庭用テレビゲーム機が挙げられる。

これら従来の家庭用テレビゲーム機は、本体をテレビに接続し、その家庭用テレビゲーム機専用のソフトウェアをROMカセットまたはCD-ROMの形で供給するソフト・ハード分離型システムを採用している。

一方、ソフト・ハード分離型の情報処理装置に対して、ソフト・ハード一体型の情報処理装置が存在する。これらの具体例としては、ソフト・ハード一体型のテレビゲーム機が挙げられる。これは前述のソフト・ハード分離型ゲーム機の出現以前に多く見られた形態である。

しかしながら、これらの情報処理装置は、その使用およびソフトウェアの開発において、それぞれ次のような問題を有するものであった。

まず、ソフト・ハード分離型の情報処理装置については、そのソフトウェアの使用に際して、利用者は前提としてプラットフォームである情報処理装置本体を入手しておく必要があった。

このことは、一方でソフト供給者にとってみれば、ソフトの販売数がプラットフォーム普及台数以上にはならないことを意味するものであり、販売対象となる

母数が、プラットフォームを有している者の数に限定されることとなる。

また、たとえばプラットフォームを家庭用テレビゲーム機とした場合は、利用者はこのプラットフォームをテレビゲームの専用装置としてとらえるため、たとえば、株価チャートや各種データベースソフトといったビジネスソフトや、英会話練習ソフトといった教育ソフトなどのゲーム以外のソフトは、たとえ販売されたとしても利用者に受け入れられにくい。また、ソフトウェアの流通システムがハードウェアのそれに依存するため、たとえばゲーム機をプラットフォームとするゲーム以外のソフトの流通と販売を行うことは極めて難しい。したがって、ソフト供給者にとって、自由なソフトの開発や販売の妨げとなっていた。

さらに、基本的には、プラットフォームに付随しているマンマシンインタフェースにあわせてソフトウェアを開発しなければならず、ソフトウェアに最適なマンマシンインタフェースを備えることは困難であった。

一方、ソフト・ハード一体型の情報処理装置については、あらかじめソフトを組み込んだ状態で販売されるため、販売母数がプラットフォーム普及台数に限定されないという利点を有する。

ここで、前述のソフト・ハード一体型のテレビゲーム機を例にとると、これらはハードウェアまで含めて専用に設計されており、マイクロプロセサを用いずにワイヤードロジックのみで構成されているものも多く存在した。

すなわち、従来のソフト・ハード一体型のテレビゲーム機においては、きわめて単純なソフトウェアしか搭載されず、また、毎回異なるハードウェアを用いていたため、ソフトウェアをモジュール化し、その再利用を行う必要はなかった。しかしながら、このような方式では、複雑なソフトウェアには対応することができない。

ところで、ソフト・ハード分離型の情報処理装置は、オペレーティングシステム、デバイスドライバ、基本入出力システムなどを有していることが多い。これは、ソフトウェア開発の効率化のみならず、異なるハードウェアシステムにおけるソフトウェアの互換性の維持を目的としたものである。アプリケーションソフトウェアの開発者は、これらのシステムモジュールを自作する必要はないが、これらを除くアプリケーションソフトウェアの部分はすべて自作する必要がある。

## 発明の概要

それぞれゆえに、この発明の主たる目的は、新規な、情報処理装置(information processing apparatus)を提供することである。

この発明の他の目的は、アプリケーションソフトウェアの開発が容易な、家庭用テレビ装置に接続して用いる情報処理装置（以下、単に「情報処理装置」と略記する場合がある。）を提供することである。

この発明は、家庭用テレビ装置にビデオ信号およびオーディオ信号を出力する情報処理装置であり、マンマシンインタフェースと、半導体メモリと、情報処理手段(information processor)とを備える。

マンマシンインタフェースは、人間がこのマンマシンインタフェースに対して与える押圧力、空間中の移動、音声の情報のうち1または複数を電気信号に変換する。

半導体メモリは情報処理手段を駆動するソフトウェアを格納し、そのソフトウェアは、オペレーティングシステム、情報処理手段内ハードウェアドライバ(information processor hardware driver)、マンマシンインタフェースドライバ、アプリケーションソフトウェアエンジンおよびアプリケーションソフトウェアコンテナ部を含む。

オペレーティングシステムは、少なくとも、このソフトウェアに含まれる全てのタスクの状態制御、タスクスケジューリング、タスク間の共有資源管理および割り込み制御を司る。

情報処理手段内ハードウェアドライバは、情報処理手段内のハードウェア資源を効率的に扱うためのものであり、ドライバプログラムとドライバデータとを含む。ドライバプログラムは、計1以上のタスクおよび／またはサブルーチンを含み、アプリケーションソフトウェアエンジンよりタスクの実行もしくはサブルーチンコールによりその機能を利用されるものであり、ドライバデータは、ドライバプログラムによって扱われるデータの集合である。

マンマシンインタフェースドライバは、マンマシンインタフェースからの電気信号を効率的にアプリケーションソフトウェアエンジンに伝達するものであり、計1以上のタスクおよび／またはサブルーチンを含み、アプリケーションソフト

ウェアエンジンよりタスクの実行もしくはサブルーチンコールによりその機能を利用される。

アプリケーションソフトウェアエンジンは、アプリケーションソフトウェアコンテンツ部が必要とする定型的な処理のうち、アプリケーションの種類に依存する処理を行うものであって、計1以上のタスクおよび／またはサブルーチンを含み、後述のアプリケーションソフトウェアコンテンツプログラムよりタスクの実行もしくはサブルーチンコールによりその機能を利用される。アプリケーションソフトウェアコンテンツ部は、アプリケーションソフトウェアコンテンツプログラムと、アプリケーションソフトウェアコンテンツデータとからなり、アプリケーションソフトウェアコンテンツプログラムは、情報処理装置の目的を達成するための特有の処理を行うプログラムコードであって、1以上のタスクを含み、アプリケーションソフトウェアコンテンツデータは、アプリケーションソフトウェアコンテンツプログラムもしくはアプリケーションソフトウェアエンジンによって扱われるデータの集合であ。

情報処理手段は、マンマシンインタフェースからの電気信号および半導体メモリに格納されているソフトウェアに基づいて、必要な演算処理を行い、映像情報および音声情報を生成する。

この発明は、第1に、ソフトウェア、ハードウェアおよびマンマシンインタフェースを一体化することにより、ソフト・ハード分離型の情報処理装置が抱える多くの問題を解決した。すなわち、利用者は、この発明の情報処理装置のみを購入すればその情報処理装置の提供するソフトウェアを使用することができ、別途プラットフォームを購入する必要がない。一方、ソフトウェアの供給者にとっては、ソフトウェアの販売がプラットフォームの普及台数に制限を受けることがないばかりでなく、自由なジャンルのソフトウェア開発を行うことができる。また、そのソフトウェアにより好適なマンマシンインタフェースを利用することについて何ら制約を受けない。

この発明は、次に、ソフト・ハードの資源をモジュール化するために、これらの境界であるインタフェースをより明確に定義付けした点に特徴を有する。

すなわち、この発明においては、一例として、従来明確にされていなかったア

アプリケーションソフトウェアエンジンという概念を新たに導入し、これをアプリケーションソフトウェアが必要とする定型的な処理のうち、アプリケーションの種類に依存する処理を行うものとしてモジュール化した。そして、情報処理装置の目的を達成するための特有の処理を行うプログラムやデータであるアプリケーションソフトウェアコンテンツ部と、アプリケーションソフトウェアエンジンとの間のインタフェースを明確に定義したものである。これによって、同種のアプリケーションの開発においては、アプリケーションソフトウェアエンジンは共通して利用できることから、アプリケーションソフトウェア開発の労力および時間が軽減される。

また、この発明に従った情報処理装置は家庭用テレビ装置に接続して用いられるため、ビデオ信号とオーディオ信号の双方を出力することが必要である。そのため情報処理装置は、映像情報を生成するプロセサと音声情報を生成するプロセサを、専用のハードウェアとして備えたものであることが好ましい。また、これらのプロセサが共有メモリ空間からのデータ取得を行う機能を備えることによって、全体の処理の高速化と回路規模の縮小が可能となる。この発明はこの点に着目し、次の構造を採用することによってこれを達成した。

詳しく述べると、この発明の実施例では、情報処理手段は、中央プロセサ(central processing unit)、グラフィックプロセサ(graphics processor)およびサウンドプロセサ(sound processor)を含み、これらのプロセサはメモリ空間を共有し、そのメモリ空間に半導体メモリが配置される。中央プロセサは、マンマシンインタフェースからの電気信号と、ソフトウェア内のプログラムコードとに基づいて、グラフィックプロセサおよびサウンドプロセサを制御し、グラフィックプロセサは、映像情報を発生する手段を備え、サウンドプロセサは、音声情報を発生する手段を備える。

また、この発明の情報処理装置のアプリケーションソフトウェアコンテンツ部は、スクリプト言語記述により構成することができる。ここで、スクリプト言語とは、情報処理手段が直接解釈可能なオブジェクトコードと比較して、抽象度が高く、人間が理解しやすい書式と構造を有する言語をいう。この場合の情報処理装置のソフトウェアの構成は、オペレーティングシステム、情報処理手段内ハ

ードウェアドライバ、マンマシンインタフェースドライバ、スクリプト言語インタプリタおよびアプリケーションソフトウェアコンテンツ部から構成される。

つまり、この発明の或る局面では、情報処理装置は、マンマシンインタフェース、半導体メモリおよび情報処理手段を備えており、マンマシンインタフェースは、人間がそのマンマシンインタフェースに対して与える押圧力、空間中の移動および音声の情報のうち 1 または複数を電気信号に変換し、半導体メモリは情報処理手段を駆動するソフトウェアを格納する。

ソフトウェアは、オペレーティングシステム、情報処理手段内ハードウェアドライバ、マンマシンインタフェースドライバ、スクリプト言語インタプリタおよびアプリケーションソフトウェアコンテンツ部を含む。

オペレーティングシステムは、少なくとも、このソフトウェアに含まれる全てのタスクの状態制御、タスクスケジューリング、タスク間の共有資源管理および割り込み制御を司る。

情報処理手段内ハードウェアドライバは、情報処理手段内のハードウェア資源を効率的に扱うためのものであり、ドライバプログラムとドライバデータとからなり、ドライバプログラムは、計 1 以上のタスクおよび／またはサブルーチンを含み、スクリプト言語インタプリタよりタスクの実行もしくはサブルーチンコールによりその機能を利用されるものであり、ドライバデータは、ドライバプログラムによって扱われるデータの集合である。

マンマシンインタフェースドライバは、マンマシンインタフェースからの情報ないし電気信号を効率的にスクリプト言語インタプリタに伝達するものであり、計 1 以上のタスクおよび／またはサブルーチンを含み、スクリプト言語インタプリタよりタスクの実行もしくはサブルーチンコールによりその機能を利用される。

スクリプト言語インタプリタは、スクリプト言語ソースコード（後述）を逐次解釈して、情報処理手段が解釈可能なオブジェクトコードを生成する。

アプリケーションソフトウェアコンテンツ部は、スクリプト言語ソースコードと、アプリケーションソフトウェアコンテンツデータとからなり、スクリプト言語ソースコードは、情報処理装置の目的を達成するための特有の処理を行うプログラムであり、アプリケーションソフトウェアコンテンツデータは、スクリプ

ト言語ソースコードもしくはスクリプト言語インタプリタによって扱われるデータの集合である。

情報処理手段は、マンマシンインタフェースからの情報ないし電気信号と、半導体メモリに格納されているソフトウェアとに基づいて演算を実行し、映像情報および音声情報を生成する。

この発明によれば、まず、マンマシンインタフェース、ソフトウェアおよびハードウェアを一体化した情報処理装置であるため、ソフトウェアの販売がハードウェアの普及台数に制限されず、また、ソフトウェアに最も適した入力装置を備えることができる。

また、ソフトウェアおよびハードウェアを含む各モジュール間のインタフェースを明確に定義付けたため、効率的でスピーディなアプリケーション開発が可能となるとともに、新たにアプリケーションソフトウェアエンジンという概念を導入したことによって同種アプリケーション開発の効率化が期待できる。

さらに、情報処理手段内ハードウェアドライバおよびマンマシンインタフェースドライバは、情報処理装置の備えるハードウェア（マンマシンインタフェースを含む）のみを制御するためのものであり、異なる機種間での互換性を保つための仕組みや、新しい機器のドライバをインストールするための仕組みを備える必要がないばかりか、情報処理手段内ハードウェアドライバおよびマンマシンインタフェースドライバのプログラムサイズを小さくでき、したがって、メモリ資源を節約できる。

さらに、実施例で用いられている高速プロセサを使用することにより、高速プロセサ内の各プロセサが大きなローカルメモリを有することなく、半導体メモリを効率的に共有可能となる。これにより、小さな回路規模で情報処理手段を実現することができ、情報処理装置の低コスト化と小型化が達成される。

この発明の上述の目的、その他の目的、特徴、および利点は、図面を参照して行う以下の実施例の詳細な説明から一層明らかとなろう。

### 図面の簡単な説明

図1はこの発明の一実施例を示すブロック図であり；



図 2 は図 1 実施例のエンジンの構成の一例を示す図解図であり；

図 3 はこの発明の他の実施例を示すブロック図であり；

図 4 は実施例の情報処理装置のソフトウェアおよびハードウェアのモジュール構成を示す機能ブロック図であり；

図 5 は高速プロセサオペレーティングシステムによって制御されるタスクの状態遷移を示す図解図であり；

図 6 はスクリプト言語を使用した場合の一実施例を示すブロック図であり；

図 7 はスクリプト言語を使用した場合の他の実施例を示すブロック図であり；

図 8 はスクリプト言語を使用した場合のモジュール構成を示す機能ブロック図であり；

図 9 は第 1 実施例であるベースボールゲーム装置のシステム構成を示す図解図であり；

図 10 はベースボールゲーム装置のゲーム画面の表示の一例を示す図解図であり；

図 11 はベースボールゲーム装置の電氣的構成を示すブロック図であり；

図 12 は高速プロセサの電氣的構成を示すブロック図であり；

図 13 はベースボールゲーム装置のモジュール構成の概略を示す機能ブロック図であり；

図 14 はベースボールゲームエンジンのグラフィック処理に着目した各部の構成を示す図解図であり；

図 15 はサウンドドライバの起動からから BGM の再生までの第 1 の手順を示すフローチャートであり；

図 16 はサウンドドライバの起動からから BGM の再生までの第 2 の手順を示すフローチャートであり；

図 17 はサウンドドライバの起動からから BGM の再生までの第 3 の手順を示すフローチャートであり；

図 18 はサウンドドライバの起動からから BGM の再生までの第 4 の手順を示すフローチャートであり；

図 19 は第 2 実施例である描画装置(drawing apparatus)のシステム構成を示す

図解図であり；

図 2 0 は描画装置のメイン画面の表示の一例を示す図解図であり；

図 2 1 は描画装置の電氣的構成を示すブロック図であり；

図 2 2 は描画装置のモジュール構成の概略を示す機能ブロック図であり；

図 2 3 は描画装置エンジン(drawing apparatus engine)のグラフィック処理に着目した構成を示す機能ブロック図であり；

図 2 4 は第 3 実施例である家庭用情報端末装置(home information terminal)のシステム構成を示す図解図であり；

図 2 5 は家庭用情報端末装置の電氣的構成を示すブロック図であり；

図 2 6 は家庭用情報端末装置のモジュール構成の概略を示す機能ブロック図であり；

図 2 7 は第 4 実施例である射撃ゲーム装置のシステム構成を示す図解図であり；

図 2 8 は射撃ゲーム装置の電氣的構成を示すブロック図であり；

図 2 9 は第 5 実施例であるカラオケ装置のシステム構成を示す図解図であり；  
そして

図 3 0 はカラオケ装置の電氣的構成を示すブロック図である。

### 好ましい実施例の詳細な説明

図 1 を参照して、この実施例の情報処理装置(information processing apparatus)は、マンマシンインタフェース 1，半導体メモリ 2 および情報処理手段(information processor) 3 を含む。また、半導体メモリ 2 には、アプリケーションソフトウェアコンテンツ部（以下、「コンテンツ部(contents portion)」と略記する場合がある。） 2 1，アプリケーションソフトウェアエンジン（以下、「エンジン」と略記する場合がある。） 2 2，オペレーティングシステム 2 3，情報処理手段内ハードウェアドライバ（以下、「ハードウェアドライバ」と略記する場合がある。） 2 4 およびマンマシンインタフェースドライバ（以下、「インタフェースドライバ」と略記する場合がある。） 2 5 が格納されている。

マンマシンインタフェース 1 は、人間がこのマンマシンインタフェースに対

して与える押圧力、空間中の移動、音声の情報のうち1または複数を電気信号ないし情報に変換するものである。押圧力を電気信号に変換するものの具体例としては、キーボードやマウスに用いられるキースイッチやタブレットの入力部分等が挙げられる。同様に、空間中の移動を電気信号に変換するものとしては、マウスの移動検出部や加速度検出器等が、また、音声の情報を電気信号に変換するものとしては、マイクロフォンなどが例示できる。

なお、マンマシンインタフェース1は、キーボードのように、同一の変換手段を複数備えていてもよく、マウスのように複数の異なる変換手段を備えていてもよい。さらに、マンマシンインタフェースは、1つの装置にキーボードとマウスが設けられている場合のように、複数の異種のデバイスで構成されているものであってもよく、1つの装置に複数のキーパッドが設けられている場合のように、複数の同種のデバイスから構成されているものであってもよい。

マンマシンインタフェース1の構成について、マウスを例にとって説明する。マウスには、X軸およびY軸のそれぞれの移動の方向および移動量を検出する手段と、キースイッチとが設けられている。マウスを平面上で移動させ、キー操作を行うと、X軸およびY軸の移動の方向および移動量およびキー操作の有無が電気信号に変換される。この電気信号は情報処理手段3に伝達されるが、このときの伝達手段は特に限定されるものではなく、ケーブル接続や赤外線通信など各種の伝達手段を採用することができる。また、伝達される電気信号が情報処理手段3で使用されるために、増幅・変換される必要がある場合は、マンマシンインタフェース1内にこれらの増幅・変換のための手段が備えられていてもよい。

半導体メモリ2は、情報処理手段3を駆動するソフトウェアを格納するメモリである。また、必要に応じてソフトウェアを高速実行するための展開領域、ソフトウェアの作業領域、実行経過の一時保存、利用者の情報の保存などの用途に用いられてもよい。

また、使用に適した半導体メモリの種類としては、マスクROM、EPROM、EEPROM（フラッシュメモリを含む。）およびSRAMやDRAM等の各種RAMなどが例示できる。これらの半導体メモリはそれぞれ用途に応じた特徴を有するものであり、情報処理装置の用途に適したものが選択されるべきである。

たとえばマスクROMは大量生産の情報処理装置に用いられるのに適している。EPROMは少量生産の情報処理装置に使用されることが好ましく、また、アプリケーションの開発用途に使用されることが好ましい。EEPROMは、少量生産向けの情報処理装置、およびアプリケーションの開発用途に使用されることが好ましい。さらに、EPROMは、電氣的に記憶内容が書き換え可能であるため、電話回線などを通じてプログラムおよび／またはデータをダウンロードして使用する情報処理装置、ソフトウェアを書き換えて使用する情報処理装置および利用者の情報の保存などにも好適する。RAMはソフトウェアを高速実行するための展開領域、ソフトウェアの作業領域、実行経過の一時保存等の用途に用いられる。また、バッテリーと組み合わせて使用されることにより、たとえばソフトウェアを書き換えて使用する情報処理装置や、利用者の情報の保存やソフトウェアの途中経過の保存などの用途に好適する。

これらのメモリは物理的に単独で用いられてもよいし、複数で用いられてもよい。複数で用いられる場合は、同種のものであっても異種のものであってもよい。たとえば、書き換えによりアプリケーションソフトの更新や交換を行う場合には、コンテンツ部21のみを更新しまたは交換すればよいので、コンテンツ部21のみをフラッシュメモリに格納し、オペレーティングシステム23などの変更不要部分をマスクROMに格納することができる。

半導体メモリ2には、図1に示すように、コンテンツ部21、エンジン22、オペレーティングシステム23、ハードウェアドライバ24およびインタフェースドライバ25が格納されており、それぞれのモジュールは以下の機能を有する。

コンテンツ部21は、情報処理装置の目的を達成するための特有の処理を行うプログラムコードであるアプリケーションソフトウェアコンテンツプログラム（以下、「コンテンツプログラム」と略記する場合がある。）と、コンテンツプログラムおよび／またはエンジンによって扱われるデータの集合であるアプリケーションソフトウェアコンテンツデータ（以下、「コンテンツデータ」と略記する場合がある。）とから構成されている。

ここで、たとえ情報処理装置がベースボールゲーム装置である場合、情報処理装置の目的とは、テレビゲームとしてのベースボールゲームを利用者に提供す

ることである。この例におけるコンテンツプログラムの処理としては、たとえばゲームモードの選択処理やタイトル表示などが例示できる。また、コンテンツデータの内容としては、投手、打者、走者の画像や投手や打者の能力を定めるパラメータが例示できる。

コンテンツ部 2 1 の形態としては、コンテンツプログラムとコンテンツデータとからなるものの他に、コンテンツデータのみで構成されているものや、スクリプト言語ソースコードとコンテンツデータとから構成されているものであってもよい。

コンテンツ部 2 1 がデータのみから構成されている場合は、アプリケーション特有の処理を行うプログラムはエンジン 2 2 に含まれる。なお、コンテンツ部 2 1 がスクリプト言語プログラムとデータとから構成されている形態については後述する。

エンジン 2 2 とは、コンテンツ部 2 1 が必要とする定型的な処理のうち、アプリケーションの種類に依存する処理を行うものである。なお、エンジンは、プロセッサ上で実行されるソフトウェアによって実現されるほか、ワイヤードロジックによるハードウェアによって実現されてもよい。特に、同じ処理を繰り返し行うエンジンはハードウェアで実現されることにより、処理の高速化を図ることができる。

エンジン 2 2 の構成としては、内部に下位的なエンジンを包含する形であってもよい。図 2 は、エンジン 2 2 の具体的な一例であり、後述のベースボールゲーム装置のエンジン 2 2 に相当するベースボールゲームエンジン a 6 2 の構成を示したものである。

ベースボールゲームエンジン a 6 2 はボールゲームエンジン a 6 2 1 を内包し、さらにボールゲームエンジン a 6 2 1 は物理運動演算エンジン a 6 2 1 1, 3 D ジオメトリエンジン a 6 2 1 2, 2 D 画像加工エンジン a 6 2 1 3 およびピクセル描画エンジン a 6 2 1 4 を内包している。

ベースボールゲームエンジン a 6 2 は、後述の実施例のような 3 次元的な画像処理を施すベースボールゲームに必要とされる処理を司るものである。このエンジンの処理の具体例としては、ボールの軌跡演算、投手・野手・打者の動き制御、

アニメーション制御，基本ルール制御，サウンド制御および入力制御などが挙げられる。

ボールゲームエンジン a 6 2 1 は、3次元空間中での球体の運動の制御などの処理を司るものである。ボールゲームエンジン a 6 2 1 はたとえば、卓球，ゴルフ，サッカーなどの他の球技ゲームにも使用可能である。このエンジンの処理の具体的としては、球体の運動制御，球体のアニメーション制御などが挙げられる。

物理運動演算エンジン a 6 2 1 1 は、3次元空間における物体の運動の座標演算を司るものである。具体的には、物体の初速度と方向，衝突の有無などの情報を与えることによって、物体の運動の軌跡を演算により求めるものである。このエンジンの処理の具体例としては、放物運動演算，衝突運動演算などが挙げられる。

3Dジオメトリエンジン a 6 2 1 2 は、3次元空間中の物体が2次元画面中でどのように表示されるかの演算を行うものである。このエンジンの処理の具体例としては、視点の移動に伴う物体の座標演算，透視投影演算などが挙げられる。

2D画像加工エンジン a 6 2 1 3 は、2次元の画像データの拡大・縮小・回転・変形の演算を行う。このエンジンの処理の具体例としては、キャラクタ拡大／縮小処理，キャラクタ回転処理などが挙げられる。

ピクセル描画エンジン(pixel rendering engine) a 6 2 1 4 は、メモリに対するピクセルの描画処理を司るものである。このエンジンの処理の具体例としては、線分・幾何図形的高速描画，高速塗りつぶし処理などが挙げられる。

オペレーティングシステム 2 3 は、少なくとも、このソフトウェアに含まれる全てのタスクの状態制御，タスクスケジューリング，タスク間の共有資源管理および割り込み制御を司るものである。タスクスケジューリングのアルゴリズムは特に限定されるものではなく、到着順アルゴリズム，優先度順アルゴリズム，ラウンドロビン法などと呼ばれるいずれのアルゴリズムが用いられてもよい。タスク間の共有資源管理の方法としては、セマフォ(semaphore)を用いた手法などが例示される。また、オペレーティングシステム 2 3 はさらに、情報処理装置の起動および初期設定を司るものであることが好ましい。さらに、メモリ管理，仮想記憶システム，入出力管理，ファイル管理，ユーザインタフェース（コマンドライ

ンインタフェース、GUIなど）の提供の機能を有していてもよい。

ハードウェアドライバ24は、情報処理手段3内のハードウェア（グラフィックプロセサ、サウンドプロセサ、DMA制御プロセサなど）を効率的に制御するためのものである。また、ハードウェアを抽象化し、より高機能な機能ブロックとして、エンジン22などの上位ソフトウェアモジュールに提供する機能を有する。これらの機能により、異なるハードウェアに対してもハードウェアドライバ24を変更することで、上位ソフトウェアモジュール（コンテンツ部21、エンジン22、オペレーティングシステム23）からは同一の機能ブロックとして扱うことが可能となる。

インタフェースドライバ25は、マンマシンインタフェース1からの入力信号を処理するものである。また、上位ソフトウェアモジュール（コンテンツ部21、エンジン22、オペレーティングシステム23）から見たマンマシンインタフェースを抽象化することにより、異なるハードウェアにより実現された同種のマンマシンインタフェースを同一のものとして扱うことが可能となる。また、単に人間からの入力を受け付けるのみではなく、人間に対して振動や音などの出力を行うマンマシンインタフェース1に対しては、それらの出力のための制御も司る。

情報処理手段3は、マンマシンインタフェース1からの電気信号と、半導体メモリ2に格納されているソフトウェアとに基づいて演算処理を行い、映像情報および音声情報を生成するものである。情報処理手段3は、1つのマイクロプロセサによって実現されるものであってもよいし、映像や音声にかかわる処理をそれぞれ独立したプロセサが司るものであってもよい。

図3に示す他の実施例においては、情報処理手段3が、中央プロセサ31、グラフィックプロセサ32およびサウンドプロセサ33から構成されている点が図1とは異なっている。

この実施例の情報処理手段3においては、中央プロセサ31、グラフィックプロセサ32およびサウンドプロセサ33のそれぞれがメモリ空間を共有し、その共有されているメモリ空間に半導体メモリ2が配置されていることが好ましい。

情報処理手段3がこのような構成をとることによって、半導体メモリ2が各プロセサに効率的に共有されるため、各プロセサは大きなローカルメモリを有する

必要がない。これにより小さな回路規模で情報処理手段 3 が構成可能となり、情報処理装置の低コスト化と小型化が同時に実現される。したがって、たとえば、マウス等のマンマシンインタフェース 1 を格納する装置（以下、「マンマシンインタフェース装置」と略記する場合がある。）内に半導体メモリ 2 および情報処理手段 3 を格納することが可能となる。

中央プロセサ 3 1 は、マンマシンインタフェース 1 より出力された電気情報ないし信号と、ソフトウェア内のプログラムコードとに基づいて、グラフィックプロセサ 3 2 およびサウンドプロセサ 3 3 の制御を行う。グラフィックプロセサ 3 2 は、ソフトウェア内の画像データなどに基づいて、映像情報を生成する。サウンドプロセサ 3 3 は、ソフトウェア内の音声データなどに基づいて、音声情報を生成する。

なお、かかる 2 つのプロセサが生成する映像情報および音声情報は、情報処理装置から家庭用テレビ装置に対して出力されるビデオ信号およびオーディオ信号と同一のものであってもよい。また、2 つのプロセサが生成する映像情報および音声情報がビデオ信号およびオーディオ信号と同一のものでない場合、情報処理装置は、映像信号からビデオ信号への変換手段（図示せず）、音声信号からオーディオ信号への変換手段（図示せず）をさらに備えるものである。たとえば、グラフィックプロセサ 3 2 から映像情報として RGB 信号が出力される場合、情報処理装置は、RGB 信号からビデオ信号への変換手段を備える。

なお、情報処理手段 3 の構成としては、特開平 10-307790 号公報（対応のアメリカ特許第 号）に開示の高速プロセサ (high-speed processor) が使用されることが好ましい。

また、グラフィックプロセサ 3 2 の構成としては、特開平 10-222151 号公報（対応のアメリカ特許第 号）に開示の走査型画像生成回路手段が使用されることが好ましい。

また、映像情報からビデオ信号への変換手段、あるいはグラフィックプロセサ 3 2 の一部として、特開平 10-301552 号公報（対応のアメリカ特許第 号）に開示のカラー・ビデオ・エンコーダが使用されることが好ましい。



図4は実施例の情報処理装置のソフトウェアおよびハードウェアのモジュール構成を示す機能ブロック図である。なお、図4以降の各図において白抜きの矢印はデータアクセスを示し、矢印は制御信号の伝達などを示す。

図4において、コンテンツ部21は、コンテンツプログラム211とコンテンツデータ212とから構成されている。また、ハードウェアドライバ24は、ドライバプログラム241とドライバデータ242とから構成されている。

コンテンツプログラム211は、情報処理装置の目的を達成するための特有の処理を行うプログラムコードであって、1以上のタスクを含むものである。また、コンテンツデータ212は、コンテンツプログラム211もしくはエンジン22によって扱われるデータの集合である。

ドライバプログラム241は、計1以上のタスクとサブルーチンを含み、エンジン22よりタスクの実行もしくはサブルーチンコールによりその機能を利用されるものであり、ドライバデータ242は、ドライバプログラム241によって扱われるデータの集合である。

情報処理装置においては、全てのタスクの状態制御はオペレーティングシステム23により行われる。すなわち、コンテンツプログラム211、エンジン22、ドライバプログラム241およびマンマシンインタフェースドライバ25に存在する全てのタスクの状態は、オペレーティングシステム23の管理下におかれている。

コンテンツプログラム211は、必要に応じてコンテンツデータ212内のデータにアクセスを行う。また、エンジン22の提供する機能を、サブルーチンコールまたはタスク生成によって利用する。

エンジン22は、必要に応じてコンテンツデータ212内のデータにアクセスを行う。また、ドライバプログラム241およびインタフェースドライバ25が提供する機能を、サブルーチンコールまたはタスク生成によって利用する。

ドライバプログラム241は、情報処理手段内ハードウェア3に対し、制御とステータスのリードとを行う。また、必要に応じてドライバデータ242にアクセスを行う。このアクセスの例としては、ドライバデータ242に楽器の波形データが含まれる場合に、ドライバプログラム241が楽器の音声再生のために行

うアクセスが挙げられる。

マンマシンインタフェースドライバ25は、電気信号に変換された人間からの入力情報を、マンマシンインタフェース1より受け取る。また、マンマシンインタフェース1が人間に対する出力を行う機能を備えている場合、出力のための制御をあわせて行う。

次にオペレーティングシステム23が司るタスクの状態制御について説明する。ここでは、後述する図13等にした高速プロセサオペレーティングシステム63におけるタスクの状態制御を例として挙げる。

高速プロセサオペレーティングシステム63は、複数のタスクの状態をイベント（事象）の発生に基づいて制御するイベント駆動型のマルチタスクオペレーティングシステムである。また、高速プロセサオペレーティングシステム63は、先に実行可能な条件が整ったタスクから順次実行していく到着順アルゴリズムに基づくタスクスケジューリングを行う。

制御されるタスクの状態には、図5に示すように、実行状態(running state)51、実行可能状態(ready state)52、待機状態(waiting state)53、停止状態54の4つが存在し、全てのタスクはこれらの状態のいずれか1つにある。実行状態51とは、タスクに後述の図12に示した中央プロセサ93の使用権が与えられ、中央プロセサ93にて処理が行われている状態である。中央プロセサ93を1つのみ備える高速プロセサ9においては、実行状態51にあるタスクは常に1以下である。実行可能状態52とは、タスクが実行されるための条件が整っているが、他のタスクが実行状態51にあるために中央プロセサ93の使用権が与えられていない状態である。この状態においては、タスクは中央プロセサ93の使用権の取得を待つ待ち行列内にある。待機状態53とは、タスクがあるイベント（事象）を待機している状態であり、そのイベントが発生するまでは中央プロセサ93の使用権を必要としない状態である。停止状態54とは、タスクが生成されていない状態、あるいはタスクの行う処理が完了し停止している状態である。

ここで、イベントの例としては、ダイレクトメモリアクセス（DMA）によるデータ転送の完了、割り込みの発生、共有されている資源の解放等が挙げられる。

図5のタスクの状態遷移について説明する。まず、全てのタスクは、生成され

るまでは停止状態 5 4 にある（初期状態）。タスクの生成は、該他のタスクによるタスク生成のシステムコール、もしくは高速プロセサオペレーティングシステム 6 3 自身により行われ、生成されたタスクは実行可能状態 5 2 へ遷移する。実行可能状態 5 2 に遷移したタスクは、中央プロセサ 9 3 の使用权の取得を待っている待ち行列の最後尾に配置される。実行状態 5 1 にあったタスクが他の状態に遷移すると、高速プロセサオペレーティングシステム 6 3 は、待ち行列の先頭にあるタスクを実行状態 5 1 へと遷移させる。これをオペレーティングシステムのディスパッチングと呼ぶ。実行状態 5 1 にあるタスクがあるイベントを待機する必要が生じた場合、該タスクはイベント待機のシステムコールを発行し、高速プロセサオペレーティングシステム 6 3 は該タスクを待機状態 5 3 へ遷移させる。待機状態 5 3 にあるタスクが待っているイベントが発生した場合、高速プロセサオペレーティングシステム 6 3 は該タスクを実行可能状態 5 2 へと遷移させる。実行可能状態 5 2 に遷移したタスクは、同様待ち行列の最後尾に配置される。実行状態 5 1 にあるタスクが処理を完了すると、タスク終了のシステムコールを高速プロセサオペレーティングシステム 6 3 に対して発行する。これを受けて、高速プロセサオペレーティングシステム 6 3 はタスクを停止状態 5 4 へと遷移させる。停止状態 5 4 へと遷移したタスクは、再び生成されるまでは停止状態 5 4 を維持する。

このように、高速プロセサオペレーティングシステム 6 3 は、イベントの待機と発生に基づいて実行状態 5 1 にあるタスクを切り替える。したがって、実行状態 5 1 にあるタスク自身がシステムコールによって実行状態 5 1 から他の状態へ遷移しない限りは、該タスクは中央プロセサ 9 3 の使用权を占有し続けることになる。これは、映像情報の処理やサウンドの再生処理などのリアルタイム処理を行うタスクが長期間中央プロセサ 9 3 の使用权を取得できない可能性を生むものである。そこで、高速プロセサオペレーティングシステム 6 3 は、実行状態 5 1 にあるタスクが自ら他のタスクに実行権を譲渡するためのシステムコールを備えている。

実行状態 5 1 にあるタスクが実行権譲渡のシステムコールを発行すると、高速プロセサオペレーティングシステム 6 3 は該タスクを実行可能状態 5 2 へと遷移

させ、実行待ち行列の最後尾へと配置する。この際、実行可能状態 5 2 にあるタスクが他に存在しなければ、該タスクは即座に実行状態 5 1 へと再び遷移し実行を継続する。

情報処理装置のアプリケーションソフトの作成において、スクリプト言語を使用することもできる。この場合の実施例が図 6 および図 7 に示される。

この場合、半導体メモリ 2 のソフトウェアは、コンテンツ部 2 1 ʼ、スクリプト言語インタプリタ（以下、「インタプリタ」と略記する場合がある。） 2 6、オペレーティングシステム 2 3、ハードウェアドライバ 2 4 およびインタフェースドライバ 2 5 とから構成されている。

図 8 はスクリプト言語を使用した実施例のモジュール構成図である。この図 8 においては、コンテンツ部 2 1 ʼ は、スクリプト言語で記述されたスクリプト言語ソースコード部 2 1 1 ʼ とコンテンツデータ 2 1 2 ʼ とから構成されている。スクリプト言語ソースコード部 2 1 1 ʼ は、情報処理装置の目的を達成するための特有の処理を行うプログラムである。具体的には、スクリプト言語ソースコード部 2 1 1 ʼ が行う処理には、アプリケーションソフトウェアが必要とする処理のうち、オペレーティングシステム 2 3、ハードウェアドライバ 2 4 およびインタフェースドライバ 2 5 が行わない処理がすべて含まれている。一方、コンテンツデータ 2 1 2 ʼ は、スクリプト言語ソースコード部 2 1 1 ʼ もしくはスクリプト言語インタプリタ 2 6 によって扱われるデータの集合である。具体的には、画像データ、音楽スコアデータ、効果音波形データおよびプログラムが用いる各種テーブルが例示される。

スクリプト言語インタプリタ 2 6 は、スクリプト言語ソースコード 2 1 1 ʼ を逐次解釈して、情報処理手段 3 が解釈可能なオブジェクトコードを生成し実行する。使用されるスクリプト言語の種類は特に限定されるものではなく、ハイパーテキストマークアップ言語（以下、「HTML」と略記）などの既存のスクリプト言語であってもよい。ただし、インタプリタ方式によるソフトウェアの実行はオーバーヘッドが大きいと、情報処理手段 3 のハードウェアに最適化された専用のスクリプト言語が用いられることが好ましい。

この実施例においても、情報処理手段 3 は、中央プロセサ 3 1 と、グラフィッ

クプロセサ 3 2 と、サウンドプロセサ 3 3 を有し、それぞれのプロセサがメモリ空間を共有し、その共有されているメモリ空間に半導体メモリ 2 が配置されている構成をとることが好ましい。

以下、具体的なこの発明の実施例を説明するが、この発明はこれらの実施例のみに限定されるべきものではない。

## 第 1 実施例

第 1 実施例はこの発明に基づいて構成された情報処理装置で構成されるベースボールゲーム装置である。

ベースボールゲーム装置は、ユーザないしプレイヤが実際に行う投球動作やバッティング動作を電気信号に変換し、この電気信号と半導体メモリ 2 に格納されているソフトウェアとに基づいて高速プロセサ 9 で演算処理を実行し、家庭用テレビ装置にビデオ信号およびオーディオ信号を出力するものである。

図 9 を参照して、この実施例のベースボールゲーム装置は、バット型入力装置 a 1、ボール型入力装置 a 2 および本体 a 3 とから構成されている。

プレイヤは、AV ケーブル 8 1 を用いて本体 a 3 と家庭用テレビ装置とを接続して用いる。バット型入力装置 a 1 の電源は、バット型入力装置 a 1 に格納された乾電池（図示せず）により供給される。ボール型入力装置 a 2 および本体 a 3 の電源は、本体 a 3 に格納された乾電池（図示せず）、あるいは本体 a 3 に接続される AC アダプタ 8 2 により供給される。

プレイヤは、バット型入力装置 a 1 を用いて実際に行うバッティング動作、ボール型入力装置 a 2 を用いて実際に行う投球動作、およびボール型入力装置 a 2 に設けられているキースイッチ a 2 2 の操作によって入力を行う。バット型入力装置 a 1 には赤外発光 LED a 1 2 が設けられており、本体 a 3 との間で赤外線による通信を行う。一方、ボール型入力装置 a 2 は、本体 a 3 とケーブルで接続されている。ベースボールゲーム装置のプレイヤは 1 人または 2 人であり、1 人で遊ぶ場合は、これらの入力装置の内の一方を使用しゲームを行う。2 人で遊ぶ場合は、これらの入力装置をそれぞれが 1 つ使用しゲームを行う。

実施例のベースボールゲーム装置では、図 1 0 に示すゲーム画面が表示される。この表示例においては、画面は打者の視点から見た球場、投手 a 4 1、野手

a 4 2 などが表示されている。また、画面上には球速 a 4 4, ゲームのスコア a 4 5, アウト, ストライク, ボールの各種カウント a 4 6, 各塁のランナの有無 a 4 7 等が表示されている。

このベースボールゲーム装置においては、一方のプレイヤが実際に行う投球動作から生じる加速度が検出され、画面上の投手 a 4 1 が、その加速度の大きさに伴ってボールのスピードやコースを変えてピッチングを行う。投手 a 4 1 が投球すると、画面上に表示されているボール a 4 3 は投手 a 4 1 の位置からホームベース a 4 8 の位置まで打者の視点から見える映像として大きさを変えて移動する。その際、投手側のプレイヤは、球種をボール型入力装置 a 2 のキースイッチ a 2 2 により選択することができる。また、打者側プレイヤがバット型入力装置 a 1 をスイングすることにより、打者がスイングのタイミングにあわせて、ボール a 4 3 を打ち返す。

図 1 1 を参照して、バット型入力装置 a 1 は、加速度検出スイッチ a 1 1 および赤外発光 L E D a 1 2 から構成されている。ボール型入力装置 a 2 は、X-Y 加速度検出器 a 2 1, ロータリーエンコーダ a 2 3 およびキースイッチセット a 2 2 から構成されている。本体 a 3 は、高速プロセサ 9, R O M a 3 2 および赤外線受光部 a 3 1 から構成されている。また高速プロセサ 9 は水晶振動子 9 7 などにより構成される発振回路を必要とする。また、この実施例においては、高速プロセサ 9 の内部メモリ 9 6 の一部を成す S R A M のデータをバックアップするため、バッテリー 9 8 が備えられている。

ボール型入力装置 a 2 とバット型入力装置 a 1 s はマンマシンインタフェース 1 に、R O M a 3 2 は半導体メモリ 2 に、高速プロセサ 9 は情報処理手段 3 にそれぞれ相当する。

バット型入力装置 a 2 は、プレイヤのスイング動作によりこの入力装置自体が空間中を移動する際の加速度を検出し、検出された加速度を電気信号に変換するものである。プレイヤが行うスイング動作により、加速度検出スイッチ a 1 1 が作動し、検出された加速度はオン／オフの 2 値を示す信号に変換される。この信号は赤外発光 L E D a 1 2 と本体 a 3 の赤外線受光部 a 3 1 との間の赤外線通信により、高速プロセサの入出力制御回路 9 1 に伝達される。入出力制御回路 9 1

は、中央プロセサ 9 の制御に基づき、信号を中央プロセサ 9 3 に伝達する。

ボール型入力装置 a 2 は、プレイヤーの投球動作によりこの入力装置自体が受ける加速度と、プレイヤーのキー操作とを電気信号に変換するものである。プレイヤーの投球動作により生じた加速度は、X-Y 加速度検出器 a 2 1 により、X-Y の直交する 2 軸の加速度を表す信号として分解されてそれぞれ検出される。検出された信号はロータリーエンコーダ a 2 3 により X 軸および Y 軸それぞれの加速度の大きさと方向を表すデジタルデータに変換され、高速プロセサ 9 の入出力制御回路 9 1 に伝達される。また、プレイヤーからのキー入力、キースイッチセット a 2 2 によりオン/オフの 2 値を示す信号に変換され、高速プロセサ 9 の入出力制御回路 9 1 に伝達される。入出力制御回路 9 1 は、中央プロセサ 9 3 の制御に基づき、の加速度の大きさと方向を表すデジタルデータと、のキースイッチセット a 2 2 からの信号とを、中央プロセサ 9 3 に伝達する。

ROM a 3 2 は、高速プロセサ 9 を駆動するソフトウェアの各モジュールを格納している。

高速プロセサ 9 は、先に挙げた特開平 1 0 - 3 0 7 7 9 0 に開示されているものと同様のものである。この高速プロセサを使用することにより、小さい回路規模にて高性能のシステムを実現することができる。

図 1 2 を参照して、高速プロセサ 9 内のグラフィックプロセサ 9 4 は、先に挙げた特開平 1 0 - 2 2 2 1 5 1 に開示されている走査型画像生成回路手段と先に挙げた特開平 1 0 - 3 0 1 5 5 2 に開示されているカラー・ビデオ・エンコーダより構成されている。

図 1 3 はこの実施例のベースボールゲーム装置のモジュール構成の概略を示す図解図である。この図 1 2 は高速プロセサ 9 内の中央プロセサ 9 3 で実行されるソフトウェアから見た、ソフトウェアおよびハードウェアを含む全てのモジュールの構成を示している。

ベースボールゲームアプリケーションプログラム a 6 1 1 およびベースボールアプリケーションデータ a 6 1 2 は、コンテンツ部 2 1 に相当する。つまり、ベースボールゲームアプリケーションプログラム a 6 1 1 はコンテンツプログラム 2 1 1 に、ベースボールアプリケーションデータ a 6 1 2 はコンテンツデータ 2

1 2にそれぞれ相当する。ベースボールアプリケーションデータ a 6 1 2 は、パラメータテーブル a 6 1 2 1, 画像データテーブル a 6 1 2 2, 画像データ a 6 1 2 3, ミュージックスコアデータテーブル a 6 1 2 4, ミュージックスコアデータ a 6 1 2 5, 効果音データテーブル a 6 1 2 6, 効果音データ a 6 1 2 7 から構成される。

ベースボールゲームエンジン a 6 2 は、エンジン 2 2 に相当する。高速プロセッサオペレーティングシステム 6 3 は、オペレーティングシステム 2 3 に相当する。

スプライトドライバ 6 4 1, テキストスクリーンドライバ 6 4 2, その他のグラフィック処理ドライバ 6 4 3, サウンドドライバ 6 4 4, インストゥルメントドライバ 6 4 5, インストゥルメントデータ 6 4 6, その他のハードウェアドライバ 6 4 7 は、ハードウェアドライバ 2 4 に相当する。これらの内インストゥルメントデータ 6 4 6 は、ドライバデータ 2 4 2 に相当し、それ以外のモジュールはドライバプログラム 2 4 1 に相当する。

ベースボールゲームインタフェースドライバ a 6 5 は、インタフェースドライバ 2 5 に相当する。

グラフィックプロセッサ 9 4, サウンドプロセッサ 9 5, 入出力制御回路 9 1 およびその他のハードウェア 9 9 は、高速プロセッサ 9 に含まれるハードウェアのモジュールである。また、描画用メモリ空間(memory apace for pixel rendering) a 9 6 は、高速プロセッサ 9 の内部メモリ 9 6 の空間の一部である。

コンテンツ部 2 1, エンジン 2 2, オペレーティングシステム 2 3, ハードウェアドライバ 2 4 およびインタフェースドライバ 2 5 に相当する全てのモジュールは、予め ROM a 3 2 に格納されている。ただし、ソフトウェアの高速実行を行うために、これらのモジュールに含まれるプログラムコードの一部は、高速プロセッサ 9 の内部メモリ 9 6 に展開される。

ベースボールゲームアプリケーションプログラム a 6 1 1 は、全体の動作制御、タイトル表示、ゲームモードの選択、各ゲームモード共通の処理、各ゲームモード特有の処理などを司る。ベースボールゲーム装置は、試合モード、フリーバッティングモード、フリーピッチングモードおよびホームランコンテストモードの 4 つのモードを備えている。ここで、各ゲームモード共通の処理として、ボール



の軌道演算などが例示できる。また、各ゲームモード特有の処理として、試合モードにおける野手のコントロールなどの処理、フリーピッチングモードにおける投手の視点からの画像処理などが例示できる。

パラメータテーブル a 6 1 2 1 は、演算を行う際にパラメータに代入される値の集合であり、例としては選手の能力データやなどが挙げられる。パラメータテーブル a 6 1 2 1 は、ベースボールゲームアプリケーションプログラム a 6 1 1 および／またはベースボールゲームエンジン a 6 2 からアクセスされる。

画像データテーブル a 6 1 2 2 は、画像データ a 6 1 2 3 の格納場所を示すテーブルである。画像データテーブル a 6 1 2 2 はベースボールゲームアプリケーションプログラム a 6 1 1 および／またはベースボールゲームエンジン a 6 2 からアクセスされ、グラフィックプロセサ 9 4 に画像データ a 6 1 2 3 の格納場所を知らせるために用いられる。

画像データ a 6 1 2 3 は、ベースボールゲーム装置で使用する画像データの集合である。球場やスコア表示部 a 4 5 などの静止画データや、ボール a 4 3, 投手 a 4 1, 野手 a 4 2 などの動画データなどが含まれる。画像データ a 6 1 2 3 には、グラフィックプロセサ 9 4 から直接アクセスされる部分と、ベースボールゲームエンジン a 6 2 が画像の拡大・縮小・回転を行う際に原画像として用いる部分の双方が含まれる。

ミュージックスコアデータテーブル a 6 1 2 4 は、ミュージックスコアデータ a 6 1 2 5 の格納場所を示すテーブルである。ミュージックスコアデータテーブル a 6 1 2 4 はベースボールゲームアプリケーションプログラム a 6 1 1 および／またはベースボールゲームエンジン a 6 2 からアクセスされ、サウンドドライバ 6 4 4 にミュージックスコアデータ a 6 1 2 5 の格納場所を知らせるために用いられる。

ミュージックスコアデータ a 6 1 2 5 は、ベースボールゲーム装置で 사용되는楽曲のミュージックスコアデータの集合である。ミュージックスコアデータ a 6 1 2 5 は、サウンドドライバ 6 4 4 からアクセスされる。

効果音データテーブル a 6 1 2 6 は、効果音データ a 6 1 2 7 の格納場所を示すテーブルである。効果音データテーブル a 6 1 2 6 はベースボールアプリケー

ションプログラム a 6 1 1 および／またはベースボールゲームエンジン a 6 2 からアクセスされ、インストゥルメントドライバ 6 4 5 およびサウンドプロセサ 9 5 に効果音データ a 6 1 2 7 の格納場所を知らせるために用いられる。

効果音データ a 6 1 2 7 は、ゲーム中に用いられる効果音などの波形データ、エンベロープデータなどの集合である。効果音の例としては、ボールがバットに当たった時の打撃音、アンパイアの音声などが挙げられる。効果音データ a 6 1 2 7 内の波形データはサウンドプロセサ 9 5 から、エンベロープデータなどはインストゥルメントドライバ 6 4 5 および／またはサウンドプロセサ 9 5 からそれぞれアクセスされる。

ベースボールゲームエンジン a 6 2 は、ベースボールゲーム装置にて必要とされる定型的な処理の内、この実施例のような 3 次元的な画像処理が行われるベースボールゲームに特有の処理を司るものである。具体的な処理の例としては、ボール軌跡演算、投手や野手の動き制御、アニメーション制御、基本ルール制御、サウンド制御、入力制御などが挙げられる。ベースボールゲームエンジン a 6 2 は、ベースボールゲームアプリケーションプログラム a 6 1 1 より、サブルーチンコールまたはタスク生成によりその機能を利用される。

高速プロセサオペレーティングシステム 6 3 は、全てのタスクの状態制御、タスクスケジューリング、タスク間の共有資源管理、割込み制御、システムの起動と初期設定などを司る。高速プロセサオペレーティングシステム 6 3 は、ベースボールゲームアプリケーションプログラム a 6 1 1、ベースボールゲームエンジン a 6 2、スプライトドライバ 6 4 1、テキストスクリーンドライバ 6 4 2、その他のグラフィック処理ドライバ 6 4 3、サウンドドライバ 6 4 4、その他のハードウェアドライバ 6 4 7、ベースボールゲームインタフェースドライバ a 6 5 に含まれる全てのタスクの状態制御を司る。タスクの状態制御の方式については図 5 で説明した通りである。また、タスクスケジューリングとしては到着順アルゴリズムを用いている。タスク間の共有資源管理にははセマフォを用いた手法を用いている。

スプライトドライバ 6 4 1 は、スプライト座標制御、スプライト番号割り当て制御、スプライト表示優先順位制御、スプライトカラー制御、可変サイズスプラ

イト表示（複数スプライトの組み合わせ）、スプライトデータ転送制御などの処理を司る。ここで、スプライトとは、ピクセルの2次元配列からなり、画面上で自由に再配置が可能な画像要素である。スプライトは動画などを表示するのに適している。スプライトドライバ641は、グラフィックプロセサ94に備えられている制御レジスタと、グラフィックプロセサ94に含まれるローカルメモリであるスプライトメモリとを介してグラフィックプロセサ94の制御を行う。また、ベースボールゲームエンジンa62より、サブルーチンコールまたはタスク生成によりその機能を利用される。

テキストスクリーンドライバ642は、テキストスクリーンの座標オフセット制御、キャラクタ優先順位制御、キャラクタ書き換え、キャラクタカラー制御などの処理を司る。ここで、キャラクタとは、ピクセルの2次元配列からなる画像要素である。テキストスクリーンは、キャラクタの2次元配列からなる画像要素であり、静止画などを表示するのに適している。テキストスクリーンドライバ642は、グラフィックプロセサ94に備えられている制御レジスタと、高速プロセサ9の内部メモリ96におかれるテキストアレイデータとを介してグラフィックプロセサ94の制御を行う。また、ベースボールゲームエンジンa62より、サブルーチンコールまたはタスク生成によりその機能を利用される。

その他のグラフィック処理ドライバ643は、ウィンドウマスク制御、HVカウンタIRQ制御などの処理を司る。ここでウィンドウマスクとは、スプライトとテキストスクリーンが合成された画像の上に画像効果を与える領域を示すものである。また、HVカウンタとは、グラフィックプロセサ94に含まれるハードウェアのカウンタであり、現在処理が行われているピクセルの水平位置と垂直位置とを指し示すものである。HVカウンタIRQとは、かかるHVカウンタが指し示す位置が予め任意に定められた位置と一致したときに中央プロセサ93に対して発生するIRQである。その他のグラフィック処理ドライバ643は、グラフィックプロセサ94に備えられている制御レジスタを介してグラフィックプロセサ94の制御を行う。また、ベースボールゲームエンジンa62より、サブルーチンコールまたはタスク生成によりその機能を利用される。

サウンドドライバ644は、アプリケーションソフトウェアコンテンツ部21

に格納されたミュージックスコアデータ a 6 1 2 5 の解釈、インストゥルメントドライバ 6 4 5 への音声チャンネルの割り当て、音声チャンネル毎のボリュームの制御などを司るものである。サウンドドライバ 6 4 4 は、サウンドプロセサ 9 5 に備えられている制御レジスタと、高速プロセサ 9 の内部メモリ 9 6 におかれるパラメータテーブルとを介してサウンドプロセサ 9 5 の制御を行う。また、ベースボールゲームエンジン a 6 2 より、サブルーチンコールまたはタスク生成によりその機能を利用される。

インストゥルメントドライバ 6 4 5 は、楽器および効果音の再生を制御するものである。1つの楽器の再生を行う毎に、1つのインストゥルメントドライバが必要とされる。インストゥルメントドライバ 6 4 5 は、インストゥルメントデータ 6 4 6 に含まれる楽器の波形データおよびエンベロープデータ、効果音データ a 6 1 2 7 に含まれる効果音の波形データおよびエンベロープデータにアクセスを行い、サウンドプロセサ 9 5 に備えられている制御レジスタと、高速プロセサ 9 の内部メモリ 9 6 におかれるパラメータテーブルとを介してサウンドプロセサ 9 5 の制御を行う。また、サウンドドライバ 6 4 4 より、サブルーチンコールによりその機能を利用される。

その他のハードウェアドライバ 6 4 7 は、DMA制御プロセサ 9 9 b、タイマ回路 9 9 a、第1バス調停回路 9 9 c および第2バス調停回路 9 9 d の制御を司るものである。その他のハードウェアドライバ 6 4 7 は、DMA制御プロセサ 9 9 a、タイマ回路 9 9 b、第1バス調停回路 9 9 c および第2バス調停回路 9 9 d にそれぞれ備えられている制御レジスタを介してこれらの各ハードウェアの制御を行う。また、ベースボールゲームエンジン a 6 2 より、サブルーチンコールまたはタスク生成によりその機能を利用される。

ベースボールゲームインタフェースドライバ a 6 5 は、マンマシンインタフェース 1 であるボール型入力装置 a 2 およびバット型入力装置 a 3 からの入力を、高速プロセサ 9 内の入出力制御回路 9 1 を通じて受け取り、ベースボールゲームエンジン a 6 2 へと伝達するものである。ここでベースボールゲームインタフェースドライバ a 6 5 は、バット型入力装置 a 1 およびボール型入力装置 a 2 のハードウェアを抽象化し、これらの入力装置をより簡便に扱うためのプログラムイ

インタフェースをベースボールゲームエンジン a 6 2 に提供する。ベースボールゲームインタフェースドライバ a 6 5 は、ベースボールゲームエンジン a 6 2 より、サブルーチンコールまたはタスク生成によりその機能を利用される。

描画用メモリ空間 a 9 6 は、拡大・縮小・回転の処理が施された画像が格納される空間であり、高速プロセッサ 9 内の内部メモリ 9 6 の一部が用いられている。描画用メモリ空間 a 9 6 への書き込みはベースボールゲームエンジン a 6 2 に内包されるピクセル描画(pixel rendering)エンジン a 6 2 1 4 により行われ、グラフィックプロセッサ 9 4 よりスプライトの画像データとして読み出される。

図 1 4 はベースボールゲームエンジン a 6 2 のグラフィック処理に注目した各部の構成を示す概略図である。ここでは、球場やスコア表示部 a 4 5 などテキストスクリーンを用いて表示される画像データの処理と、ボール a 4 3, 投手 a 4 1, 野手 a 4 2 などスプライト用いて表示される画像データの処理を分けて記述し、またそれぞれの処理に伴う各部の機能を併せて記述する。

先ず、テキストスクリーンを用いて表示される画像データの処理と各部の機能について記述する。前述のように、ベースボールゲームエンジン a 6 2 はボールゲームエンジン a 6 2 1 を内包し、ボールゲームエンジン a 6 2 1 は物理運動演算エンジン a 6 2 1 1, 3D ジオメトリエンジン a 6 2 1 2, 2D 画像加工エンジン a 6 2 1 3, ピクセル描画エンジン a 6 2 1 4 の 4 つのエンジンを内包している。また、ベースボールゲームエンジン a 6 2 はベースボールゲームエンジン制御 a 6 2 2 を、ボールゲームエンジン a 6 2 1 はボールゲームエンジン制御 a 6 2 1 5 をそれぞれ備えている。

ベースボールゲームエンジン制御 a 6 2 2 は、球場やスコア表示 a 4 5 の画像を表示するために、画像データテーブル a 6 1 2 2 よりこれらの画像の格納位置を示す情報を読み出す。またベースボールゲームエンジン制御 a 6 2 2 は、スコア・カウント等の画面表示についての情報、それぞれの画像の格納位置を示す情報、視点の座標および方向を示す情報をボールゲームエンジン制御 a 6 2 1 5 へ伝達する。

ボールゲームエンジン制御 a 6 2 1 5 は、ベースボールゲームエンジン制御 a 6 2 2 より受け取った情報に基づいて、テキストスクリーン全体のオフセット座

標、テキストスクリーンを構成するキャラクタのアトリビュート情報などを算出し、テキストスクリーンドライバ642へ伝達する。

テキストスクリーンドライバ642は、ベースボールゲームエンジン制御a622より受け取った情報に基づいて、テキストスクリーンの座標オフセット制御、キャラクタ優先順位制御、キャラクタ書き換え、キャラクタカラー制御などの処理を司る。

グラフィックプロセサ94は、テキストスクリーンドライバ642からの制御に従って、直接表示キャラクタ画像データa61231よりテキストスクリーンを構成するキャラクタの画像データを読み出し、テキストスクリーンの画像を生成する。

次に、スプライトを用いて表示される画像データの処理と各部の機能について記述する。ベースボールゲームエンジン制御a622は、ボールや野手などの画像を拡大・縮小・回転処理の原画像データとして扱うために、画像データテーブルa6122よりこれらの画像の格納位置を示す情報を読み出す。また、スプライトを用いて表示される画像には、拡大・縮小・回転処理を施されずにグラフィックプロセサ94より画像データが直接読み出されて表示されるものも含まれるため、これらの画像の格納位置を示す情報の読み出しも行う。またベースボールゲームエンジン制御a622は、ボールの運動情報、投手、野手、打者および走者の位置および状態等の情報、視点の座標および方向の情報を算出し、ボールゲームエンジン制御a6215へ伝達する。

ボールゲームエンジン制御a6215は、ベースボールゲームエンジン制御a622より受け取った情報に基づいて、ボールの運動状態を表すパラメータの値を算出し、物理運動演算エンジンa6211へ伝達する。また、投手、野手などオブジェクトの3次元座標を算出し、視点の3次元座標および方向ベクトルの情報と共に、3Dジオメトリエンジンa6212へ伝達する。

物理運動演算エンジンa6211は、この実施例においては、ボールの投球に伴う放物運動演算、ボールとバットなどが衝突する際の衝突運動演算の処理を行う。物理運動演算エンジンa6211は、先ずボールゲームエンジン制御a6215よりボールの運動状態を表すパラメータを受け取る。このパラメータの例と

しては、投球時のボールの初速度と方向ベクトル、ボールと衝突する物体の質量と速度ベクトルなどが挙げられる。物理運動演算エンジン a 6 2 1 1 は、受け取ったパラメータとボールの現在の運動状態とに基づいてボールの運動状態の変化を演算し、次の画面描画時のボールの 3 次元座標を、ボールゲームエンジン制御 a 6 2 1 5 を介して、あるいは直接、3D ジオメトリエンジン a 6 2 1 2 へ伝達する。

3D ジオメトリエンジン a 6 2 1 2 は、この実施例においては、打者または投手の視点の変化に伴う各オブジェクトの 3 次元空間中の相対移動演算と、3 次元空間座標から 2 次元スクリーン座標への透視投影演算の処理を行う。まず 3D ジオメトリエンジン a 6 2 1 2 は、ボールの 3 次元座標を物理運動演算エンジン a 6 2 1 1 またはボールゲームエンジン制御 a 6 2 1 5 より、投手や野手などの 3 次元座標と視点の 3 次元座標および方向ベクトルをボールゲームエンジン制御 a 6 2 1 5 より受け取る。受け取ったこれらの情報に基づいて、各オブジェクトのスクリーン座標と、各オブジェクトを表すキャラクタの拡大縮小率と回転角度を求め、各オブジェクトのスクリーン座標をボールゲームエンジン制御 a 6 2 1 5 へ、キャラクタの拡大縮小率と回転角度を 2D 画像加工エンジン a 6 2 1 3 へ伝達する。

2D 画像加工エンジン a 6 2 1 3 は、この実施例においては、ボールなどを表すキャラクタに対し、拡大・縮小・回転の画像処理を施す。2D 画像加工エンジン a 6 2 1 3 は、ボールゲームエンジン制御 a 6 2 1 5 よりキャラクタの拡大縮小率と回転角度を受け取り、画像データ a 6 1 2 3 の一部である拡大縮小キャラクタ画像データ (a61232) よりキャラクタの原画像データを読み出し、拡大・縮小・回転が施された画像データを演算により求める。演算の結果をピクセル毎のキャラクタ内オフセット座標とカラー情報といった形で、ピクセル描画エンジン a 6 2 1 4 に伝達する。

ピクセル描画エンジン a 6 2 1 4 は、この実施例においては、ボールなどを表すキャラクタの描画用メモリ空間 a 9 6 への描画処理を行う。ピクセル描画エンジン a 6 2 1 4 は、2D 画像加工エンジン a 6 2 1 3 よりピクセル毎のキャラクタ内オフセット座標とカラー情報とを受け取り、各ピクセルのメモリ空間上のア

ドレスとビット位置を算出し、指定されたカラー情報の書き込みを行う。ここで、カラー情報の1ピクセル当りのビット数とメモリの1ワードのビット数が一致しない場合、ピクセル描画エンジン a 6 2 1 4 は1ピクセル単位でカラー情報を書き換えるためにリード・モディファイ・ライトのアクセスを描画用メモリ空間 a 9 6 に対して行う。

ボールゲームエンジン制御 a 6 2 1 5 は、3Dジオメトリエンジン a 6 2 1 2 より受け取った各オブジェクトのスクリーン座標に基づいて、スプライトのスクリーン座標とアトリビュート情報を算出し、スプライトドライバ 6 4 1 へ伝達する。

スプライトドライバ 6 4 1 は、ボール、投手、野手などの各オブジェクトを表示するために、スプライト座標制御、スプライト番号割り当て制御、スプライト表示優先順位制御、スプライトカラー制御、スプライトデータ転送制御の処理を司る。また、オブジェクトを表すキャラクタのサイズがスプライトのサイズより大きい場合、1つのオブジェクトを表示するために複数のスプライトが組み合わされて用いられる。スプライトドライバ 6 4 1 は、このような複数スプライトの組み合わせによる可変サイズスプライトの表示の処理も司る。

グラフィックプロセッサ 9 4 は、スプライトドライバ 6 4 1 からの制御に従って、描画用メモリ空間 a 9 6 および直接表示キャラクタ画像データ (a61231) よりスプライトの画像データを読み出し、スプライトの画像を生成する。

グラフィックプロセッサ 9 4 は、全てのテキストスクリーンとスプライトの画像を合成し、ビデオ信号を生成し出力する。

次に、ソフトウェアおよびハードウェアの各モジュールがどのように機能しているのかを、バックグラウンドミュージック（以降BGMと略記）の再生を例にとり、各モジュールの処理を示したフローチャートを用いて説明する。図 1 5, 図 1 6, 図 1 7 および図 1 8 は、サウンドドライバの起動からBGMの再生までの手順を示したフローチャートである。

これらのフローチャートに示される処理や判断などのステップは、それぞれの最上段に記載のモジュールによって行われ、たとえば、「サウンドドライバの起動」(Se2) の処理はベースボールゲームエンジン a 6 2 により行われる処理であ



ることを示している。同様に「サウンドドライバの初期設定」(Sd1)の処理はサウンドドライバ644により行われる処理である。なお、図15、図16、図17および図18の各処理等の中にはタスクの状態を表すもの(St1, St2など)等のステップではないものが含まれており、これらは中央プロセサ93にて行われる処理を表すものではない。中央プロセサ93の実行は実線の矢印に従って行われる。また、図中では、ベースボールアプリケーションプログラムa611はアプリケーションプログラムと略記されている。

図15の場合、ベースボールゲーム装置が起動すると、まず高速プロセサオペレーティングシステム63が実行を開始し、ハードウェアおよびソフトウェアの基本的な初期設定を行う。ここで行われる初期設定は、高速プロセサ9で実行されるソフトウェア全てに共通して必要とされるものであり、アプリケーションの種類に依存しないものである。その後、高速プロセサオペレーティングシステム63は、第1に実行されることを予め指定されているタスクの生成を行う。このタスクはベースボールゲームエンジンa62に含まれている。

ベースボールゲームエンジンa62は、まずアプリケーションにとって必要な初期設定を行う必要があり、この初期設定にはサウンドドライバの起動(Se2)が含まれる。ベースボールゲームエンジンa62は、サウンドドライバ644を起動する際に、サウンドドライバ644の使用するスタックエリアの指定、テンポイベントの指定、サウンドドライバ644に割り当てるスロット数の設定を行う(Se2)。ここで、テンポイベントとは、サウンドドライバ644の処理の時間単位を定めるイベントであり、中央プロセサ93に対するマスク不能割り込み(以降NMIと略記することがある)、タイマ回路99aからの割り込み要求(以降タイマIRQと略記することがある)等の一定間隔で発生するイベントであることが望ましい。高速プロセサ9においては、NMIはビデオ信号の垂直ブランキング期間の開始時にグラフィックプロセサ94より発行されるものである。また、ここでスロットとは、サウンドプロセサ95が再生を制御している音声チャンネルのことを指す。サウンドプロセサ95は、同時に16の音声チャンネルの再生が可能である。

サウンドドライバ644の起動は、サウンドドライバ644に含まれるサウン

ドドライバ644の初期設定を行うためのサブルーチンをコールすることで行われる。この初期設定には、スタックエリアの設定、割り当てられたスロット数の設定、割り当てられた全スロットの初期化等の処理が含まれる（Sd1）。初期化終了後、サウンドドライバ644は、テンポイベントに基づいてサウンド再生の処理を司るタスクの生成を行う（Sd2）。ここでは、このタスクをサウンドドライバタスクと呼称している。

ところで、あるタスクが他のタスクを生成する場合、タスクの生成は高速プロセサオペレーティングシステム63に対するシステムコールの発行によって行われる。ベースボールゲームエンジンa62のタスクから呼び出されたサウンドドライバ644のサブルーチンは、サウンドドライバタスクを生成するシステムコールを高速プロセサオペレーティングシステム63に発行し、高速プロセサオペレーティングシステム63はこのシステムコールを受けてサウンドドライバタスクの生成処理を行い（So2）、サウンドドライバタスクを停止状態から実行可能状態（St1）へと遷移させる（So2）。これは、高速プロセサオペレーティングシステム63がサウンドドライバタスクを待ち行列の最後尾に配置することを意味する。

高速プロセサオペレーティングシステム63のタスク生成処理が終了すると、サウンドドライバ644に中央プロセサ93の実行が戻る。

その後、サウンドドライバ644はサブルーチンの終了を行い、中央プロセサ93の実行はベースボールゲームエンジンa62へと戻り、サブルーチンコール直後の処理から実行が継続される。

ベースボールゲームエンジンa62はテンポイベントの生成のシステムコールを発行し（Se3）、これを受けて高速プロセサオペレーティングシステム63はテンポイベントの生成処理を行う（So3）。

これをもって、サウンドドライバ644の起動が終了し、ベースボールゲームエンジンa62の処理は、アプリケーションが必要とする他の初期設定へと移行する（Se4）。

図16の場合、実行状態にあったタスクが、他の状態への遷移を行うシステムコールを発行すると、高速プロセサオペレーティングシステム63はタスクの切

替処理（So5）を行い、実行可能状態にあるタスクの中から待ち行列の先頭にあるタスクを実行状態へと遷移させる。ここで、サウンドドライバタスクが待ち行列の先頭であれば、高速プロセサオペレーティングシステム63はサウンドドライバタスクを実行状態（St2）へ、他のタスクが待ち行列の先頭であればそのタスクを実行状態へ遷移させる（So8）。

サウンドドライバタスクはテンポイベントに基づいて処理を行うため、実行状態になったサウンドドライバタスクは、テンポイベント待ちのシステムコールを発行する（St3）。これを受けて高速プロセサオペレーティングシステム63はサウンドドライバタスクを待機状態へ遷移させ（So10）、他のタスクを実行状態にする（So11）。

図17の場合、テンポイベントは、ベースボールゲームエンジンa62に管理されている。テンポイベントが発生すると（Se6）、ベースボールゲームエンジンa62はテンポイベント発生 of システムコールを発行し、テンポイベントの通過を高速プロセサオペレーティングシステム63に知らせる（So12）。これを受けて高速プロセサオペレーティングシステム63は、サウンドドライバタスクを待機状態から実行可能状態に遷移させ、待ち行列の最後尾に配置する（So13）。

図18の場合、ベースボールゲームアプリケーションプログラムa611は、BGM再生開始の指示をベースボールゲームエンジンa62内のサブルーチンをコールすることで行う（Sa2）。サブルーチンコールの際に、再生するBGMのナンバを入力引数としてベースボールゲームエンジンa62に渡す。この例では、BGM#1の再生開始の指示が行われている。

ベースボールゲームエンジンa62は、BGM#1再生開始の指示をベースボールゲームアプリケーションプログラムa611より受けると、サウンドドライバ644内のサブルーチンをコールし、BGM#1のスコアデータの先頭アドレスと、サウンドドライバ644がBGM#1を再生するために用いるワークエリアの指定とを入力引数としてサウンドドライバ644に渡す（Se7）。

コールされたサウンドドライバ644のサブルーチンは、BGM#1の再生開始をエントリに設定（Sd3）した後、サブルーチンを終了する。ここで中央プロセサ93の実行は、ベースボールゲームエンジンa62に戻る。その後、ベース

ボールゲームエンジン a 6 2 はサブルーチンを終了し、中央プロセサ 9 3 の実行は、ベースボールゲームアプリケーションプログラム a 6 1 1 へ戻る。

中央プロセサ 9 3 の実行がベースボールゲームアプリケーションプログラム a 6 1 1 に戻ると、ベースボールゲームアプリケーションプログラム a 6 1 1 は次の処理へ進むことができる (Sa3)。

サウンドドライバタスクがテンポイベントの通過を受けて実行状態になると、まずエントリのチェックを行う (St7)。ここでエントリが設定されていない場合、サウンドドライバタスクは今回の処理を終了し、次のテンポイベントを待機するシステムコールを発行し、待機状態へ遷移する。

BGM再生開始のエントリを確認する、あるいはBGMが再生中であることを確認すると、サウンドドライバタスクは、スロット管理の処理を行う (St9)。まず今回の再生に使用するスロット数を確認し、空きスロットの数がそれを下回っている場合、再生状態にあるスロットの中で優先順位の低いものから新たに再生用のスロットとして割り当てる。ここで、優先順位の低いものとは、まず残響音などの再生を行っているスロットがこれに当たる。楽器の再生を行っているスロットの中では、早くに再生を開始したものから低い優先順位が設定される。

その後、サウンドドライバタスクは、設定されているスコアデータのポインタに基づいてスコアデータの読み出しを行い (St10)、読み出されたスコアデータを解釈する。

解釈されたスコアデータにしたがい、サウンドドライバ 6 4 4 は、今回の処理にて再生が行われるインストゥルメントドライバ 6 4 5 を順次コールしてゆく。インストゥルメントドライバ 6 4 5 は、サブルーチンとしてその機能をサウンドドライバ 6 4 4 に提供する。

サウンドドライバ 6 4 4 は、インストゥルメントドライバ # 1 をコールする際に、インストゥルメントドライバ # 1 に割り当てるスロットの指定、再生の残りイベント数の指定、ベロシティの指定などを入力引数としてインストゥルメントドライバ # 1 に渡す (St121)。ここでベロシティとは、たとえばピアノの鍵盤を叩く強さ、ギター弦を弾く強さなどを表すものであり、最終的には再生される波形と各音声チャンネルのボリュームに反映されるものである。

コールされたインストゥルメントドライバ# 1は、入力引数に従ってサウンドプロセサ9 5の制御を行う。サウンドプロセサ9 5の制御は、サウンドプロセサ9 5に内蔵される制御レジスタと、内部メモリ9 6を介して受け渡されるパラメータテーブルとを通じて行われる。インストゥルメントドライバ# 1は、サウンドプロセサ9 5に対し、自身に割り当てられたスロットの制御、再生モードの指定、チャンネルボリュームの指定などを行う (Si1)。

サウンドプロセサ9 5は、インストゥルメントドライバ6 4 5からの制御に従い、音声チャンネルの再生を行う (Sp1, Sp2, . . . , SpN)。

サウンドドライバ6 4 4は、同様にインストゥルメントドライバ# 2, . . . , # Nの再生を行うか否かの判断 (St112, . . . , St11N) と、再生を行う場合は各インストゥルメントドライバの制御 (St122, . . . , St12N) とを行う。

サウンドドライバ6 4 4は、今回のイベントの処理にて再生が行われる全てのインストゥルメントドライバのコールを終えると、スコアデータのポインタを更新し (St13)、次回のテンポイベントを待機するシステムコールを発行し、待機状態へ遷移する。

上述の手順に従えば、ベースボールゲームアプリケーションプログラム a 6 1 1のBGM再生に関する処理は再生開始指示を行うだけでよい。すなわち、簡易な処理で音楽を再生することが可能となることから、プログラムの作成が極めて容易となる。

## 第2実施例

実施例2はユーザがマウスを操作することによって家庭用テレビ装置の画面に描画(drawing)を行うためのものである。図19を参照して、この実施例の描画装置は、本体b 1および電源供給のための付属装置b 2から構成される。

ユーザは付属装置b 2をAVケーブル8 1を用いて家庭用テレビ装置と接続し、本体b 1と付属装置b 2をケーブル接続することにより本装置を用いる。電源は、付属装置b 2に接続されるACアダプタ8 2により供給される。

ユーザは、マウスとして機能する本体b 1自体を机などの平板上で滑らせる移動動作、および本体b 1に設けられた2つのキースイッチb 1 1 1の操作によって入力を行う。

図20に示す表示例においては、メイン画面には描画エリアb41と、フレームb42と、各種アイコンb43が表示されている。また、本体b1自体の移動に対応して画面上を動くマウスポインタb46が表示されている。

描画装置は、予め本体b1内のROMb12に格納されている背景画の画像データを描画エリアb41に展開する機能、描画エリアb41にステッカを貼り付ける機能、ステッカを動画で表現することができるアニメーションステッカ機能などを補助機能として有している。

ステッカは画面の左端のステッカアイコンb44に表示されている。ステッカアイコンb44の上部に配置される切替アイコンb45は、ステッカアイコンb44の一覧を切り替えるためのものである。ユーザはステッカアイコンb44をマウスポインタb46でドラッグすることによって、描画エリアb41にステッカを貼り付けることができる。

また、実施例の描画装置の特徴として、描画エリアb41にステッカを貼りつける前に、そのステッカを自由に拡大・縮小・回転・フリップさせることができる。フリップとは画像を鏡に写したように反転させることであり、描画装置においては水平方向または垂直方向のフリップが可能である。

図21はこの実施例の電氣的構成を示すブロック図である。本体b1は、マウス入力デバイスb11、高速プロセサ9、ROMb12および外部SRAMB13から構成されている。また、マウス入力デバイスb11は、X-Y位置移動検出器b112、ロータリーエンコーダb113およびキースイッチセットb111から構成されている。高速プロセサ9は水晶振動子97などにより構成される発振回路を必要とする。また、この実施例においては、高速プロセサ9の内部メモリ96の一部を成すSRAMと外部SRAMB13のデータをバックアップするため、バッテリー98が備えられている。

ここで、マウス入力デバイスb11はマンマシンインタフェース1に、ROMb12と外部SRAMB13は半導体メモリ2に、高速プロセサ9は情報処理手段3にそれぞれ相当する。

マウス入力デバイスb11は、ユーザが行う本体b1自体の移動とキースイッチb111によるキー操作とを検出し、電気信号に変換するものである。ユーザ

が本体b 1 自体を机などの平板上で移動させる動作は、X－Y位置移動検出器b 1 1 2により、X－Yの直交する2軸の移動量を表す信号として検出される。検出された信号はロータリーエンコーダb 1 1 3によりX軸、Y軸それぞれの移動方向と移動量を表すデジタルデータに変換され、高速プロセサ9の入出力制御回路9 1に伝達される。また、ユーザからのキー入力、キースイッチセットb 1 1 1によりオン／オフの2値を示す信号に変換され、高速プロセサ9の入出力制御回路9 1に伝達される。入出力制御回路9 1は、中央プロセサ9 3の制御に基づき、移動の方向と移動量を表すデジタルデータと、キースイッチセットb 1 1 1からの信号とを、中央プロセサ9 3に伝達する。

ROMb 1 2は、高速プロセサ9を駆動するソフトウェアの各モジュールを格納している。外部SRAMB 1 3は、描画エリアb 4 1の画像データの格納と保存、ステッカの拡大縮小回転のための作業領域などに用いられる。ROMb 1 2および外部SRAMB 1 3は、いずれも高速プロセサ9の外部メモリとして、高速プロセサ9の外部バスに接続されている。

高速プロセサ9は、先の特開平1 0－3 0 7 7 9 0号公報に開示されているものと同様のものである。この高速プロセサを使用することにより、小さい回路規模にて高性能のシステムを実現できることから、単一の装置内にマンマシンインタフェース1と半導体メモリ2と情報処理手段3とを格納することが可能となった。

次に、図2 2にはこの実施例の描画装置のソフトウェア、ハードウェアを含むモジュールの構造が示されるが、この図2 2は高速プロセサ9内の中央プロセサ9 3で実行されるソフトウェアから見た、ソフトウェアおよびハードウェアを含む全てのモジュールの構成を示している。

図2 2における描画装置アプリケーションプログラムb 6 1 1および描画装置アプリケーションデータb 6 1 2は、コンテンツ部2 1に相当する。ここで、描画装置アプリケーションプログラムb 6 1 1はコンテンツプログラム2 1 1に、描画装置アプリケーションデータb 6 1 2はコンテンツデータ2 1 2にそれぞれ相当する。描画装置アプリケーションデータb 6 1 2は、パラメータテーブルb 6 1 2 1、画像データテーブルb 6 1 2 2、画像データb 6 1 2 3、ミュージ

クスコアデータテーブルb 6 1 2 4, ミュージックスコアデータb 6 1 2 5, 効果音データテーブルb 6 1 2 6, 効果音データb 6 1 2 7から構成されている。

描画装置エンジンb 6 2は、エンジン2 2に相当する。高速プロセサオペレーティングシステム6 3は、オペレーティングシステム2 3に相当する。

スプライトドライバ6 4 1, テキストスクリーンドライバ6 4 2, その他のグラフィック処理ドライバ6 4 3, サウンドドライバ6 4 4, インストゥルメントドライバ6 4 5, インストゥルメントデータ6 4 6, その他のハードウェアドライバ6 4 7は、ハードウェアドライバ2 4に相当する。これらの内インストゥルメントデータ6 4 6は、ドライバデータ2 4 2に相当し、それ以外のモジュールはドライバプログラム2 4 1に相当する。

マウスインタフェースドライバb 9 5は、インタフェースドライバ2 5に相当する。グラフィックプロセサ9 4と、サウンドプロセサ9 5と、入出力制御回路9 1と、その他のハードウェア9 9は、高速プロセサ9に含まれるハードウェアのモジュールである。また、描画用メモリ空間b 9 6は、外部S R A M b 1 3の空間の一部である。

コンテンツ部2 1, エンジン2 2, オペレーティングシステム2 3, ハードウェアドライバ2 4, インタフェースドライバ2 5に相当する全てのモジュールは、予めR O M b 1 2に格納されている。但しソフトウェアの高速実行を行うために、これらのモジュールに含まれるプログラムコードの一部は、高速プロセサ9の内部メモリ9 6に展開される。

描画装置アプリケーションプログラムb 6 1 1は、全体の動作制御, タイトル表示, 描画モードの選択, 描画データの保存, 観賞モードの実行等の処理を司る。ここで、描画装置における描画モードには、背景画が色付きであるモードと線画のみのモード, 背景画像が消去可能であるモードと消去不可のモードがあり、これらの組み合わせにより計4つの描画モードが用意されている。

パラメータテーブルb 6 1 2 1は、演算を行う際にパラメータに代入される値の集合であり、例としてはアニメーションパターンを示すデータなどが挙げられる。パラメータテーブルb 6 1 2 1は、描画装置アプリケーションプログラムb 6 1 1および/または描画装置エンジンb 6 2からアクセスされる。



画像データテーブルb 6 1 2 2は、画像データb 6 1 2 3の格納場所を示すテーブルである。画像データテーブルb 6 1 2 2は描画装置アプリケーションプログラムb 6 1 1および／または描画装置エンジンb 6 2からアクセスされ、グラフィックプロセサ9 4に画像データb 6 1 2 3の格納場所を知らせるために用いられる。

画像データb 6 1 2 3は、描画装置で使用する画像データの集合である。背景画、フレーム、アイコン、マウスポインタ、ステッカなどの画像データが含まれる。画像データb 6 1 2 3には、グラフィックプロセサ9 4から直接アクセスされる部分と、描画装置エンジンb 6 2がステッカの拡大・縮小・回転を行う際に原画像として用いる部分の双方が含まれる。

ミュージックスコアデータテーブルb 6 1 2 4は、ミュージックスコアデータb 6 1 2 5の格納場所を示すテーブルである。ミュージックスコアデータテーブルb 6 1 2 4は描画装置アプリケーションプログラムb 6 1 1および／または描画装置エンジンb 6 2からアクセスされ、サウンドドライバ6 4 4にミュージックスコアデータb 6 1 2 5の格納場所を知らせるために用いられる。

ミュージックスコアデータb 6 1 2 5は、描画装置のソフトウェアで 사용되는楽曲のミュージックスコアデータの集合である。ミュージックスコアデータb 6 1 2 5は、サウンドドライバ6 4 4からアクセスされる。

効果音データテーブルb 6 1 2 6は、効果音データb 6 1 2 7の格納場所を示すテーブルである。効果音データテーブルb 6 1 2 6は描画装置アプリケーションプログラムb 6 1 1および／または描画装置エンジンb 6 2からアクセスされ、インストゥルメントドライバ6 4 5およびサウンドプロセサ9 5に効果音データb 6 1 2 7の格納場所を知らせるために用いられる。

効果音データb 6 1 2 7は、描画装置のソフトウェア中で用いられる効果音などの波形データ、エンベロープデータなどの集合である。効果音の例としては、描画の際の効果音やアイコンの選択時の効果音が挙げられる。効果音データb 6 1 2 7内の波形データはサウンドプロセサ9 5から、エンベロープデータなどはインストゥルメントドライバ6 4 5および／またはサウンドプロセサ9 5からそれぞれアクセスされる。

描画装置エンジンb 6 2は、描画装置のソフトウェアにて必要とされる定型的な処理の内、描画装置のソフトウェアに特有の処理を司るものである。具体的な処理の例としては、幾何図形描画、ステッカ加工、動画アニメーション制御、サウンド制御、入力制御などが挙げられる。描画装置エンジンb 6 2は、描画装置アプリケーションプログラムb 6 1 1より、サブルーチンコールまたはタスク生成によりその機能を利用される。

高速プロセサオペレーティングシステム6 3は、第1実施例に用いられているものと同じものが用いられている。高速プロセサオペレーティングシステム6 3は、描画装置アプリケーションプログラムb 6 1 1、描画装置エンジンa 6 2、スプライトドライバ6 4 1、テキストスクリーンドライバ6 4 2、その他のグラフィック処理ドライバ6 4 3、サウンドドライバ6 4 4、その他のハードウェアドライバ6 4 7、マウスインタフェースドライバb 9 5に含まれる全てのタスクの状態制御を司る。

ハードウェアドライバ2 4に相当する各モジュールは、第1実施例に用いられているものと同じものが用いられている。ただし、これらのモジュールの内、スプライトドライバ6 4 1、テキストスクリーンドライバ6 4 2、その他のグラフィック処理ドライバ6 4 3、サウンドドライバ6 4 4、その他のハードウェアドライバ6 4 7の各機能は、描画装置エンジンb 6 2からサブルーチンコールまたはタスク生成により利用されるものである。

マウスインタフェースドライバb 9 5は、マンマシンインタフェース1であるマウス入力デバイスb 1 1からの入力を、高速プロセサ9内の入出力制御回路9 1を通じて受け取り、描画装置エンジンb 6 2へと伝達するものである。ここでマウスインタフェースドライバb 9 5は、マウス入力デバイスb 1 1のハードウェアを抽象化し、この入力手段をより簡便に扱うためのプログラムインタフェースを描画装置エンジンb 6 2に提供する。

外部S R A M b 1 3は、拡大・縮小・回転の処理が施されたステッカの画像データを格納するための領域と、描画エリアの画像データの格納と保存のための領域とを備える。外部S R A M b 1 3への書き込みは描画装置エンジンb 6 2により行われる。描画エリアへの貼り付けがなされる前のステッカの画像データはス

プライトとして、描画エリアの画像データはテキストスクリーンとして、グラフィックプロセッサ94よりそれぞれ読み出される。

図23は描画装置エンジンb62のグラフィック処理に着目した各部の構成を示す概略図である。図23に記載の通り、描画装置エンジンb62は、アニメーションエンジンb624、幾何図形描画エンジンb622、2D画像加工エンジンa6214、ピクセル描画エンジンa6213を内包している。また、描画装置エンジンb62は、描画装置エンジン制御b625を備えている。ここで、2D画像加工エンジンa6214およびピクセル描画エンジンa6213は、実施例1の2D画像加工エンジンa6213およびピクセル描画エンジンa6214と同じものである。

ここでは、背景画の処理と、アイコン、フレーム、マウスポインタおよびアニメーションステッカの処理と、ステッカの拡大・縮小・回転処理と、幾何図形描画、自由曲線描画、塗りつぶし描画の処理とを分けて記述し、またそれぞれの処理に伴う各部の機能を併せて記述する。

先ず、背景画の処理において、描画装置エンジン制御b625は、外部SRAMB13の描画用メモリ空間b96に背景画の画像データを書き込むために、画像データテーブルb6122より背景画の画像データの格納位置を示す情報を読み出す。また、読み出した格納位置を示す情報に従って、背景画の画像データを画像データ（背景画）b61231から読み出し、読み出した画像データを外部SRAMB13の描画メモリ空間b96に直接書き込む。

次に、アイコン、フレーム、マウスポインタ、アニメーションステッカの処理において、描画装置エンジン制御b625は、フレームとアイコンをテキストスクリーンのキャラクタとして表示するために、またマウスポインタとアニメーションステッカをスプライトとして表示するために、画像データテーブルb6122よりこれらの画像の格納位置を示す情報を読み出す。描画装置エンジン制御b625は、フレームおよびアイコンの種類と位置を示す情報をテキストスクリーンドライバ642へ伝達し、マウスポインタおよびアニメーションステッカの種類と座標を示す情報をアニメーションエンジンb624へ伝達する。

テキストスクリーンドライバ642は、描画装置エンジン制御b625より受

け取った情報に基づいて、テキストスクリーンの座標オフセット制御，キャラクタ優先順位制御，キャラクタ書き換え，キャラクタカラー制御などの処理を司る。

アニメーションエンジンb 6 2 4は、描画装置エンジン制御b 6 2 5より受け取った情報と、現在のマウスポインタ／アニメーションステッカのステータスとに基づいて、パラメータテーブルb 6 1 2 1に格納されているアニメーションパターンを示すデータを読み出し、次の画面更新時のマウスポインタ／アニメーションステッカのアニメーションパターンを算出し、それぞれの座標の情報と共にスプライトドライバ6 4 1へ伝達する。

スプライトドライバ6 4 1は、マウスポインタとアニメーションステッカを表示するために、スプライト座標制御，スプライト番号割り当て制御，スプライト表示優先順位制御，スプライトカラー制御，スプライトデータ転送制御，可変サイズスプライト表示の処理を司る。

グラフィックプロセサ9 4は、テキストスクリーンドライバ6 4 2およびスプライトドライバ6 4 1からの制御に従って、画像データ（アイコン，フレーム，マウスポインタ）b 6 1 2 3 2より画像データを読み出し、テキストスクリーンおよびスプライトの画像を生成する。

次に、ステッカの拡大・縮小・回転処理において、描画装置エンジン制御b 6 2 5は、マウスインタフェースドライバb 9 5を通じて受け取ったユーザからの入力に従って、ステッカの座標、拡大縮小率および回転角度を算出する。また、算出したステッカの座標をスプライトドライバ6 4 1へ、ステッカの拡大縮小率および回転角度を2 D画像加工エンジンa 6 2 1 4へ伝達する。

スプライトドライバ6 4 1は、ステッカをスプライトとして表示するために、スプライト座標制御，スプライト番号割り当て制御，スプライト表示優先順位制御，スプライトカラー制御，スプライトデータ転送制御，可変サイズスプライト表示の処理を司る。

2 D画像加工エンジンa 6 2 1 4は、描画装置エンジン制御b 6 2 5よりステッカの拡大縮小率と回転角度を受け取り、画像データ（ステッカ）（b61233）よりステッカの原画像データを読み出し、拡大・縮小・回転が施されたステッカの画像データを演算により求める。演算の結果をピクセル毎の座標とカラー情報と

いった形で、ピクセル描画エンジン a 6 2 1 3 に伝達する。

ピクセル描画エンジン a 6 2 1 3 は、2 D 画像加工エンジン a 6 2 1 4 よりピクセル毎の座標とカラー情報とを受け取り、各ピクセルのメモリ空間上のアドレスとビット位置を算出し、描画用メモリ空間 b 9 6 に指定されたカラー情報の書き込みを行う。

グラフィックプロセサ 9 4 は、スプライトドライバ 6 4 1 からの制御に従って、描画用メモリ空間 b 9 6 より拡大・縮小・回転の処理が施されたステッカの画像データを読み出し、スプライトの画像を生成する。

次に、幾何図形描画、自由曲線描画、塗りつぶし描画などの処理と各部の機能について記述する。この実施例において、幾何図形とは直線や円などの図形であり、自由曲線とはマウスポインタの軌跡をなぞるように描かれる線である。また、塗りつぶし描画とは、線分で囲まれた領域を特定の色または画像パターンで塗りつぶす処理を指す。

描画装置エンジン制御 b 6 2 5 は、マウスインタフェースドライバ b 9 5 を通じて受け取ったユーザからの入力に従って、幾何図形の種類、座標、線種、カラー情報を算出し、幾何図形描画エンジン b 6 2 2 へ伝達する。また、同様にユーザからの入力に従って、自由曲線描画／塗りつぶし描画にて描画される各ピクセルの座標とカラー情報を算出し、ピクセル描画エンジン a 6 2 1 3 へ伝達する。

幾何図形描画エンジン b 6 2 2 は、直線や円などの幾何図形を描画するための処理を行う。描画装置エンジン b 6 2 5 より受け取った幾何図形の種類、座標、線種、カラー情報から、ピクセル単位の座標とカラー情報を算出し、ピクセル描画エンジン a 6 2 1 3 へ伝達する。

ピクセル描画エンジン a 6 2 1 3 は、描画エリアの画像データの描画処理を行う。幾何図形描画エンジン b 6 2 2 より幾何図形の、描画装置エンジン制御 b 6 2 5 より自由曲線や塗りつぶし処理の、ピクセル単位の座標とカラー情報とを受け取り、各ピクセルのメモリ空間上のアドレスとビット位置を算出し、描画用メモリ空間 b 9 6 に指定されたカラー情報の書き込みを行う。

グラフィックプロセサ 9 4 は、テキストスクリーンドライバ 6 4 2 からの制御に従って、描画用メモリ空間 b 9 6 より描画エリア b 4 1 の画像データを読み出

し、テキストスクリーンの画像を生成する。

### 第3実施例

第3実施例に従った家庭用情報端末装置は、アナログ公衆回線を通じてプログラムおよびデータを送受信する機能を備え、受信したプログラムおよび／またはデータと、家庭用情報端末装置のユーザからのキー入力および音声入力に基づいて高速プロセサ9にて処理が行われ、家庭用テレビ装置にビデオ信号とオーディオ信号とを出力するものである。この端末装置は、ネットワークを用いたホームバンキングサービス、ショッピング、証券取引、ゲームなどに好適する。

図24を参照して、実施例の家庭用情報端末装置は、本体c1のみから構成される。ユーザは本体c1を、AVケーブル81を用いて家庭用テレビ装置と接続し、モジュラーケーブル83を用いて電話回線モジュラージャックと接続して用いる。電源は、本体c1に格納された乾電池(図示せず)、あるいは本体c1に接続されるACアダプタ82より供給される。また、ユーザは、キースイッチセットc11を用いて、メニューの選択やデータの入力などの操作を行い、マイクロフォンc12を用いて音声の入力を行う。

図25はこの実施例の電氣的構成を示すブロック図である。本体c1は、マイクロフォンc12、増幅回路c13、キースイッチセットc11、ROMc17、外部SRAMc18、EEPROMc19、高速プロセサ9、モデムLSIc15、回線インタフェースc14から構成される。また、高速プロセサ9とモデムLSIc15は、水晶振動子97などにより構成される発振回路をそれぞれ必要とする。

ここで、マイクロフォンc12とキースイッチセットc11はマンマシンインタフェース1に、ROMc17と外部SRAMc18とEEPROMc19は半導体メモリ2に、高速プロセサ9は情報処理手段3にそれぞれ相当する。

ユーザからの音声入力は、マイクロフォンc12によりアナログ電圧信号に変換されたのち、増幅回路c13によって電圧の振幅増幅がなされ、高速プロセサ9のA/Dコンバータ92に入力される。A/Dコンバータ92は、中央プロセサ93からの制御に基づき、入力されたアナログ電圧信号をデジタルデータに変換し、変換されたデータを中央プロセサ93に伝達する。

ユーザからのキー入力、キースイッチセットc 1 1によりオン／オフの2値を表す信号に変換され、高速プロセサ9の入出力制御回路9 1に入力される。入出力制御回路9 1は、中央プロセサ9からの制御に基づき、入力されたデータを中央プロセサ9 3に伝達する。

ROMc 1 7は、高速プロセサ9を駆動するソフトウェアに含まれるモジュールのうち、通信によって取得されないものを予め格納している。

外部SRAMc 1 8は、高速プロセサ9を駆動するソフトウェアに含まれるモジュールのうち、通信によって取得されるものを格納する。また、ソフトウェアの作業領域、実行経過の一時保存を行う用途に用いられてもよい。

EEPROMc 1 9は、電氣的にデータの書き込みと消去が可能な不揮発性の半導体メモリであり、ユーザの個人情報、取引記録、パスワードなどの情報を保持する。

ROMc 1 7, 外部SRAMc 1 8, EEPROMc 1 9は、いずれも高速プロセサ9の外部メモリとして、高速プロセサ9の外部バスに接続されている。

この実施例における高速プロセサc 1 3は、特開平1 0 - 3 0 7 7 9 0に開示されているものと同様のものである。このプロセサを使用することにより小さい回路規模にて高性能のシステムを実現できることから、単一の装置内にマンマシンインタフェース1と半導体メモリ2と情報処理手段3とを格納することが可能となった。

モデムLSIc 1 5は、アナログ公衆回線を通じてプログラムおよびデータの送受信を行うためのものである。デジタルデータを変調しアナログ信号として送信する機能と、アナログ信号を復調しデジタルデータとして受信する機能を備える。変調方式を含めた通信規格は、標準化されている既存の規格が用いられることが好ましい。また、モデムLSIc 1 5自体は、通信等を制御するためのマイクロコントローラを内蔵しているものであることが望ましい。

回線インタフェースc 1 4は、回線制御回路(NCU)としての機能と、電話線の2線-4線の変換回路と、インピーダンスの整合回路とを備える。

図2 6に家庭用情報端末装置のモジュール構成の概略を示すが、この図2 6は、高速プロセサ9内の中央プロセサ9 3で実行されるソフトウェアから見た、ソフ

トウェアおよびハードウェアを含む全てのモジュールの構成を示している。

図26のスクリプト言語ソースコードc611と、パラメータテーブルc6122と、画像データテーブルc6122と、画像データc6123と、ミュージックスコアデータテーブルc6124と、ミュージックスコアデータc6125と、効果音データテーブルc6126と、効果音データc6127は、コンテンツ部21に相当する。この内、スクリプト言語ソースコードc611は、スクリプト言語ソースコード部211に相当し、それ以外のモジュールは全てコンテンツデータ212に相当する。

スクリプト言語インタプリタc66は、スクリプト言語インタプリタ26に相当する。高速プロセサオペレーティングシステム63は、オペレーティングシステム23に相当する。

スプライトドライバ641と、テキストスクリーンドライバ642と、その他のグラフィック処理ドライバ643と、サウンドドライバ644と、インストゥルメントドライバ645と、インストゥルメントデータ646と、その他のハードウェアドライバ647は、ハードウェアドライバ24に相当する。これらの内インストゥルメントデータ646は、ドライバデータ242に相当し、それ以外のモジュールはドライバプログラム241に相当する。

音声入力ドライバ651と、キースイッチセットドライバ652は、インタフェースドライバ25に相当する。

グラフィックプロセサ94と、サウンドプロセサ95と、A/Dコンバータ92と、入出力制御回路91と、その他のハードウェア99は、高速プロセサ9に含まれるハードウェアのモジュールである。

スクリプト言語ソースコードc611は、スクリプト言語で記述されたプログラムコードであり、本装置の目的となる処理を司るものである。たとえばこの端末装置をホームバンキング装置として用いる場合、残高照会および振込みなどの基本機能、オンラインショッピングおよび家計簿機能などの拡張機能などの処理を司る。また、必要に応じて、コンテンツデータ212に相当するモジュールに対するアクセスを行う。

コンテンツデータ212に相当する各モジュールは、実施例1および2に



用いられているものと基本的に同じ構造を有している。ただし、図中に示されるモジュールを常に全て備えている必要はない。また、通信により取得したスクリプト言語ソースコード c 6 1 1 より扱われるのに適した独自の構造を有していてもよい。

コンテンツ部 2 1 に相当するモジュールは、基本的には通信により取得されるものであるが、予め ROM c 1 7 に格納されている部分があってもよい。前述のホームバンキング装置を例にとると、基本機能のための処理、メニューセレクトの処理、アプリケーションとして必要な初期設定などに用いられるプログラムおよびデータは、予め ROM c 1 7 に格納されることが望ましい。

スクリプト言語インタプリタ c 6 6 は、スクリプト言語で記述されたスクリプト言語ソースコード c 6 1 1 を、高速プロセサ 9 内の中央プロセサ 9 4 が解釈可能なオブジェクトコードに逐次変換して実行するものである。

スクリプト言語の一例として、HTML が挙げられる。本家庭用情報端末装置がインターネットのワールドワイドウェブ (WWW) からコンテンツを取得する用途に用いられる場合、スクリプト言語インタプリタ c 6 6 は HTML を解釈可能なものであるべきである。しかしながら、スクリプト言語を逐次解釈しながら実行することは処理上のオーバーヘッドが大きいため、既存のスクリプト言語で記述されたコンテンツを解釈する必要が無い場合には、高速プロセサ 9 を効率良く制御するための専用のスクリプト言語が用いられることが望ましい。

高速プロセサオペレーティングシステム 6 3 は、先の第 1 および第 2 実施例に用いられているものと同じものが用いられている。高速プロセサオペレーティングシステム 6 3 は、スクリプト言語インタプリタ c 6 6、スプライトドライバ 6 4 1、テキストスクリーンドライバ 6 4 2、その他のグラフィック処理ドライバ 6 4 3、サウンドドライバ 6 4 4、その他のハードウェアドライバ 6 4 7、音声入力ドライバ c 6 5 1、キースイッチセットドライバ c 6 5 2、通信プロトコル制御プログラム 2 7、モデムドライバ 2 8 に含まれる全てのタスクの状態制御を司る。

ハードウェアドライバ 2 4 に相当する各モジュールは、先の第 1 および第 2 実施例に用いられているものと同じものが用いられている。ただし、これらのモジ

ジュールの内、スプライトドライバ641, テキストスクリーンドライバ642, その他のグラフィック処理ドライバ643, サウンドドライバ644, その他のハードウェアドライバ247の各機能は、スクリプト言語インタプリタc66からサブルーチンコールまたはタスク生成により利用されるものである。

音声入力ドライバc651は、A/Dコンバータ92を制御し、マイクロフォンc12から入力されたアナログ電圧信号のデジタルデータへの変換を司る。また、必要に応じて変換後のデジタルデータの加工も司る。音声入力ドライバc651は、スクリプト言語インタプリタc66より、サブルーチンコールまたはタスク生成によってその機能を利用される。音声入力ドライバc651を通じて変換・加工されたデジタルデータは、スクリプト言語インタプリタc66へ伝達される。

キースイッチセットドライバc652は、キースイッチセットc11からの入力信号を入出力制御回路91を通じて読み出し、キーのステータス、キーのオン／オフのトリガ検出などの情報をスクリプト言語インタプリタc66へ伝達する。キースイッチセットドライバc11は、スクリプト言語インタプリタc66より、サブルーチンコールまたはタスク生成によりその機能を利用される。

通信プロトコル制御プログラム27は、データ通信における上位的な通信プロトコルの処理を司る。上位的な通信プロトコルの例としては、インターネットを介するデータ通信における、パケット転送プロトコル(TCP)などのトランスポート層のプロトコル、インターネットプロトコル(IP)などのインターネット層のプロトコルが例示できる。下位的な通信プロトコルの処理およびモデムLSIの制御はモデムドライバ28によって行われるため、通信プロトコル制御プログラムは、タスク生成またはサブルーチンコールによりモデムドライバ28の機能を利用する。また、通信プロトコル制御プログラム27は、スクリプト言語インタプリタc66からの送信データの受け取り、スクリプト言語インタプリタc66への受信データの受け渡しを司る。

モデムドライバ28は、モデムLSI c15の制御と、下位的な通信プロトコルの処理とを司る。下位的な通信プロトコルの例としては、インターネットを介するデータ通信における、X.25などのサブネットワーク層のプロトコル、ポ

イントラポイントプロトコル（PPP）などのリンク層のプロトコルが例示できる。モデムドライバ28は、通信プロトコル制御プログラム27からのタスク生成またはサブルーチンコールにより、その機能を利用される。

コンテンツ部21'に相当する各モジュールの内、通信によって取得されたものは、外部SRAMc18に格納される。

スクリプト言語インタプリタc66と、高速プロセサオペレーティングシステム63と、ハードウェアドライバ24に相当する各モジュール641-647と、インタフェースドライバ25に相当する各モジュールc651, c652と、通信プロトコル制御プログラム27と、モデムドライバ28は、ROMc17に格納されている。また、アプリケーションソフトウェアコンテンツ部21'に相当する各モジュールの内の一部も、予めROMc17に格納されている。

#### 第4実施例

第4実施例の射撃ゲーム装置は、利用者がライトガンを操作することによって家庭用テレビ装置の画面に表示される標的を射撃するゲームを、利用者に提供するものである。

図27第2示すように、この射撃ゲーム装置は、2つのライトガン型入力装置d1, d2および本体d3から構成される。

プレイヤは本体d3をAVケーブル81を用いて家庭用テレビ装置と接続し、2つのライトガン型入力装置d1, d2と本体d3をケーブル接続することにより本装置を用いる。電源は、本体d3に格納された乾電池（図示せず）、あるいは本体d3に接続されるACアダプタ82により供給される。

プレイヤは、ライトガン型入力装置d1, d2を用いて画面上のオブジェクトを射撃することによって、メニューの選択や標的の射撃などの操作を行う。

図28を参照して、ライトガン型入力装置d1, d2は、受光素子d12, d22およびキースイッチセットd11, d21から構成されている。本体d3は、高速プロセサ9、ROMd31から構成されている。また、高速プロセサ9は、水晶振動子97などにより構成される発振回路を必要とする。また、この実施例においては、高速プロセサ9の内部メモリ96の一部を成すSRAMのデータをバックアップするため、バッテリー98が備えられている。

ここで、2つのライトガン型入力装置d 1, d 2はマンマシンインタフェース装置1に、ROMd 3 1は半導体メモリ2に、高速プロセサ9は情報処理手段3にそれぞれ相当する。

プレイヤからのキー入力、キースイッチセットd 1 1, d 2 1によりオン／オフの2値を表す信号に変換され、高速プロセサ9の入出力制御回路9 1に入力される。入出力制御回路9 1は、中央プロセサ9 3からの制御に基づき、入力されたデータを中央プロセサ9 3に伝達する。

また、受光素子d 1 2, d 2 2は家庭用テレビ装置のCRTのスキャンにより発する光を受光し、受光したタイミングを高速プロセサ9内のグラフィックプロセサ9 4に伝達する。このタイミングでグラフィックプロセサ9 4内のHVカウンタの値がラッチされ、ラッチされた値は中央プロセサ9 3から読み出される。この一連の処理により、ライトガン型入力装置d 1, d 2の受光部がCRT上のどの位置を指しているかが検出される。

ROMd 3 1は、高速プロセサ9を駆動するソフトウェアの各モジュールを格納している。ROMd 3 1は、高速プロセサ9の外部メモリとして、高速プロセサ9の外部バスに接続されている。

この実施例における高速プロセサ9は、先の実施例と同様に、特開平10-307790号公報に開示されているものと同様のものである。このプロセサを使用することにより小さい回路規模にて高性能のシステムを実現することが可能となった。

## 第5実施例

第5実施例のカラオケ装置は、家庭用テレビ装置にビデオ信号とオーディオ信号を出力し、家庭用テレビ装置から出力される楽曲の音声と画面上の歌詞に合わせて、マイクを用いてカラオケを楽しむためのものである。

図29に示すように、この実施例のカラオケ装置は、本体e 1のみから構成される。ユーザは本体e 1を、AVケーブル8 1を用いて家庭用テレビ装置と接続して用いる。電源は、本体e 1に格納された乾電池（図示なし）より供給される。カラオケ装置のユーザは、キースイッチセットe 1 2を用いて、選曲や再生速度、キー調整などの操作を行い、マイクロフォンe 1 1を用いて音声の入力

を行う。

図30に示すように、本体e1は、マイクロフォンe11、増幅回路e13、キースイッチセットe12、ROMe14、高速プロセサ9から構成されている。また、高速プロセサ9は、水晶振動子97などにより構成される発振回路を必要とする。またこの実施例においては、高速プロセサ9の内部メモリ96の一部を成すSRAMのデータをバックアップするため、バッテリー98が備えられている。

ここで、マイクロフォンe11とキースイッチセットe12はマンマシンインタフェース1に、ROMe14は半導体メモリ2に、高速プロセサ9は情報処理手段3にそれぞれ相当する。

ユーザからの音声入力は、マイクロフォンe11によりアナログ電圧信号に変換されたのち、増幅回路e13によって電圧の振幅増幅がなされ、高速プロセサ9のA/Dコンバータ92に入力される。A/Dコンバータ92は、中央プロセサ93からの制御に基づき、入力されたアナログ電圧信号をデジタルデータに変換し、変換されたデータを中央プロセサ93に伝達する。

ユーザからのキー入力は、キースイッチセットe12によりオン/オフの2値を表す信号に変換され、高速プロセサ9の入出力制御回路91に入力される。入出力制御回路91は、中央プロセサ93からの制御に基づき、入力されたデータを中央プロセサ93に伝達する。

ROMe14は、高速プロセサ9を駆動するソフトウェアの各モジュールを格納している。ROMe14は、高速プロセサ9の外部メモリとして、高速プロセサ9の外部バスに接続されている。

この実施例における高速プロセサ9は、先の特開平10-307790号公報に開示されているものと同様のものである。このプロセサを使用することにより小さい回路規模にて高性能のシステムを実現できることから、単一の装置内にマンマシンインタフェース1と半導体メモリ2と情報処理手段3とを格納することが可能となった。

この発明が詳細に説明され図示されたが、それは単なる図解および一例として用いたものであり、限定であると解されるべきではないことは明らかであり、この発明の精神および範囲は添付されたクレームの文言によってのみ限定される。

## クレーム：

1. 家庭用テレビ装置にビデオ信号およびオーディオ信号を出力する情報処理装置であり、

マンマシンインタフェースと、半導体メモリと、情報処理手段とを備えており、  
前記マンマシンインタフェースは、人間が本マンマシンインタフェースに対して与える押圧力、空間中の移動、音声の情報のうち1または複数を電気信号に変換し、

前記半導体メモリは前記情報処理手段を駆動するソフトウェアを格納し、

前記ソフトウェアは、オペレーティングシステムと、情報処理手段内ハードウェアドライバと、マンマシンインタフェースドライバと、アプリケーションソフトウェアエンジンと、アプリケーションソフトウェアコンテンツ部とからなり、

前記オペレーティングシステムは、少なくとも、本ソフトウェアに含まれる全てのタスクの状態制御と、タスクスケジューリングと、タスク間の共有資源管理と、割り込み制御とを司り、

前記情報処理手段内ハードウェアドライバは、前記情報処理手段内のハードウェア資源を効率的に扱うためのものであり、ドライバプログラムと、ドライバデータとからなり、

前記ドライバプログラムは、計1以上のタスクとサブルーチンを含み、前記アプリケーションソフトウェアエンジンよりタスクの実行もしくはサブルーチンコールによりその機能を利用されるものであり、

前記ドライバデータは、前記ドライバプログラムによって扱われるデータの集合であり、

前記マンマシンインタフェースドライバは、前記マンマシンインタフェースからの前記電気信号を効率的に前記アプリケーションソフトウェアエンジンに伝達するものであり、計1以上のタスクとサブルーチンを含み、前記アプリケーションソフトウェアエンジンよりタスクの実行もしくはサブルーチンコールによりその機能を利用されるものであり、

前記アプリケーションソフトウェアエンジンは、前記アプリケーションソフトウェアコンテンツ部が必要とする定型的な処理のうち、アプリケーションの種類

に依存する処理を行うものであって、計1以上のタスクとサブルーチンを含み、後記アプリケーションソフトウェアコンテンツプログラムよりタスクの実行もしくはサブルーチンコールによりその機能を利用されるものであり、

前記アプリケーションソフトウェアコンテンツ部は、アプリケーションソフトウェアコンテンツプログラムと、アプリケーションソフトウェアコンテンツデータとからなり、

アプリケーションソフトウェアコンテンツプログラムは、本情報処理装置の目的を達成するための特有の処理を行うプログラムコードであって、1以上のタスクを含み、

アプリケーションソフトウェアコンテンツデータは、前記アプリケーションソフトウェアコンテンツプログラムもしくは前記アプリケーションソフトウェアエンジンによって扱われるデータの集合であり、

前記情報処理手段は、前記マンマシンインタフェースからの電気信号と、前記半導体メモリに格納されているソフトウェアとに基づいて演算処理を行い、映像情報および音声情報を生成するものである。

2. クレーム1に従属する情報処理装置であって、wherein 前記情報処理手段は、中央プロセサと、グラフィックプロセサと、サウンドプロセサとを備えており、

中央プロセサと、グラフィックプロセサと、サウンドプロセサとはメモリ空間を共有し、そのメモリ空間に前記半導体メモリが配置されており、

前記中央プロセサは、前記マンマシンインタフェースからの前記電気情報と、前記ソフトウェア内のプログラムコードとに基づいて、前記グラフィックプロセサおよび前記サウンドプロセサの制御を行い、

前記グラフィックプロセサは、映像情報を発生する手段を備え、

前記サウンドプロセサは、音声情報を発生する手段を備える。

3. 家庭用テレビ装置にビデオ信号およびオーディオ信号を出力する情報処理装置であり、

マンマシンインタフェースと、半導体メモリと、情報処理手段とを備えており、

前記マンマシンインタフェースは、人間が本マンマシンインタフェースに対し

て与える押圧力、空間中の移動、音声の情報のうち1または複数を電気信号に変換し、

前記半導体メモリは前記情報処理手段を駆動するソフトウェアを格納し、

前記ソフトウェアは、オペレーティングシステムと、情報処理手段内ハードウェアドライバと、マンマシンインタフェースドライバと、スクリプト言語インタプリタと、アプリケーションソフトウェアコンテンツ部とからなり、

前記オペレーティングシステムは、少なくとも、本ソフトウェアに含まれる全てのタスクの状態制御と、タスクスケジューリングと、タスク間の共有資源管理と、割り込み制御とを司り、

前記情報処理手段内ハードウェアドライバは、前記情報処理手段内のハードウェア資源を効率的に扱うためのものであり、ドライバプログラムと、ドライバデータとからなり、

前記ドライバプログラムは、計1以上のタスクとサブルーチンを含み、前記スクリプト言語インタプリタよりタスクの実行もしくはサブルーチンコールによりその機能を利用されるものであり、

前記ドライバデータは、前記ドライバプログラムによって扱われるデータの集合であり、

前記マンマシンインタフェースドライバは、前記マンマシンインタフェースからの前記電気情報を効率的に前記スクリプト言語インタプリタに伝達するものであり、計1以上のタスクとサブルーチンを含み、前記スクリプト言語インタプリタよりタスクの実行もしくはサブルーチンコールによりその機能を利用されるものであり、

前記スクリプト言語インタプリタは、後記スクリプト言語ソースコードを逐次解釈して、前記情報処理手段が解釈可能なオブジェクトコードを生成し実行するものであり、

前記アプリケーションソフトウェアコンテンツ部は、スクリプト言語ソースコードと、アプリケーションソフトウェアコンテンツデータとからなり、

前記スクリプト言語ソースコードは、本情報処理装置の目的を達成するための特有の処理を行うプログラムであり、



前記アプリケーションソフトウェアコンテンツデータは、前記スクリプト言語ソースコードもしくは前記スクリプト言語インタプリタによって扱われるデータの集合であり、

前記情報処理手段は、前記マンマシンインタフェースからの電気信号と、前記半導体メモリに格納されているソフトウェアとに基づいて演算処理を行い、映像情報および音声情報を生成するものである。

4. クレーム3に従属する情報処理装置であって、wherein 前記情報処理手段は、中央プロセサと、グラフィックプロセサと、サウンドプロセサとを備えており、

中央プロセサと、グラフィックプロセサと、サウンドプロセサとはメモリ空間を共有し、そのメモリ空間に前記半導体メモリが配置されており、

前記中央プロセサは、前記マンマシンインタフェースからの前記電気情報と、前記ソフトウェア内のプログラムコードとに基づいて、前記グラフィックプロセサおよび前記サウンドプロセサの制御を行い、

前記グラフィックプロセサは、映像情報を発生する手段を備え、

前記サウンドプロセサは、音声情報を発生する手段を備える。

5. クレーム1または2に従属する情報処理装置であって、further comprising 一般通信回線を通じてデータおよび／またはプログラムを送受信することができる通信手段、wherein

前記情報処理手段は、前記マンマシンインタフェースからの電気信号と、前記半導体メモリに格納されているソフトウェアに加え、さらに通信手段から得られたデータおよび／またはプログラムとに基づいて演算処理を行うものである。

6. クレーム3または4に従属する情報処理装置であって、further comprising 一般通信回線を通じてデータおよび／またはプログラムを送受信することができる通信手段、wherein

前記情報処理手段は、前記マンマシンインタフェースからの電気信号と、前記半導体メモリに格納されているソフトウェアに加え、さらに通信手段から得られたデータおよび／またはプログラムとに基づいて演算処理を行うものである。

7. クレーム1, 2および5のいずれかに従属する情報処理装置であって、

wherein 前記マンマシンインタフェース手段と、前記半導体メモリと、前記情報処理手段とは、単一の装置内に格納されている。

8. クレーム3, 4および6のいずれかから従属する情報処理装置であって、  
wherein 前記マンマシンインタフェース手段と、前記半導体メモリと、前記情報処理手段とは、単一の装置内に格納されている。